2019

CICD 持续集成

JENKINS+GITLAB

致力于明天

主机配置

ecs-nginx 41a48a73-2a5f-47d6-9...	可用区3	运行中	1vCPUs | 1GB | s3.small.1 CentOS 7.6 64bit	192.168.1.100 (私有)	按需计费	远程登录	更多
ecs-jenkins 0bb1ada5-672e-4c22-af...	可用区3	运行中	2vCPUs | 4GB | s3.large.2 CentOS 7.6 64bit	139.159.189.191 (弹性公网) ... 192.168.1.10 (私有)	按需计费	远程登录	更多
ecs-gitlab 71c9aaa4-60a2-460d-8...	可用区3	运行中	2vCPUs | 4GB | s3.large.2 CentOS 7.6 64bit	139.9.91.82 (弹性公网) 5 Mb... 192.168.1.11 (私有)	按需计费	远程登录	更多

gitlab这台机我给的是2cpu4g内存

Gitlab 环境部署

安装依赖包。

sudo yum install -y curl policycoreutils-python openssh-server

```
================================================================================
 Package                  Arch          Version                 Repository    Size
================================================================================
Installing:
 policycoreutils-python   x86_64        2.5-29.el7_6.1          updates      456 k
Installing for dependencies:
 audit-libs-python        x86_64        2.8.4-4.el7             base          76 k
 checkpolicy              x86_64        2.5-8.el7               base         295 k
 libcgroup                x86_64        0.41-20.el7             base          66 k
 libsemanage-python       x86_64        2.5-14.el7              base         113 k
 python-IPy               noarch        0.75-6.el7              base          32 k
 setools-libs             x86_64        3.3.8-4.el7             base         620 k

Transaction Summary
================================================================================
Install  1 Package (+6 Dependent packages)

Total download size: 1.6 M
Installed size: 5.3 M
Downloading packages:
warning: /var/cache/yum/x86_64/7/base/packages/audit-libs-python-2.8.4-4.el7.x86_64.rpm: Header V3 RSA/SHA
256 Signature, key ID f4a80eb5: NOKEY
Public key for audit-libs-python-2.8.4-4.el7.x86_64.rpm is not installed
(1/7): audit-libs-python-2.8.4-4.el7.x86_64.rpm                |  76 kB  00:00:00
(2/7): libcgroup-0.41-20.el7.x86_64.rpm                        |  66 kB  00:00:00
(3/7): checkpolicy-2.5-8.el7.x86_64.rpm                        | 295 kB  00:00:00
(4/7): python-IPy-0.75-6.el7.noarch.rpm                        |  32 kB  00:00:00
(5/7): libsemanage-python-2.5-14.el7.x86_64.rpm                | 113 kB  00:00:00
(6/7): setools-libs-3.3.8-4.el7.x86_64.rpm                     | 620 kB  00:00:00
Public key for policycoreutils-python-2.5-29.el7_6.1.x86_64.rpm is not installed
(7/7): policycoreutils-python-2.5-29.el7_6.1.x86_64.rpm        | 456 kB  00:00:00
--------------------------------------------------------------------------------
Total                                              2.1 MB/s | 1.6 MB  00:00:00
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
Importing GPG key 0xF4A80EB5:
 Userid     : "CentOS-7 Key (CentOS 7 Official Signing Key) <security@centos.org>"
 Fingerprint: 6341 ab27 53d7 8a78 a7c2 7bb1 24c6 a8a7 f4a8 0eb5
```

设置 SSH 开机自启动并启动 SSH 服务。

sudo systemctl enable sshd

sudo systemctl start sshd

安装 Postfix 来发送通知邮件。

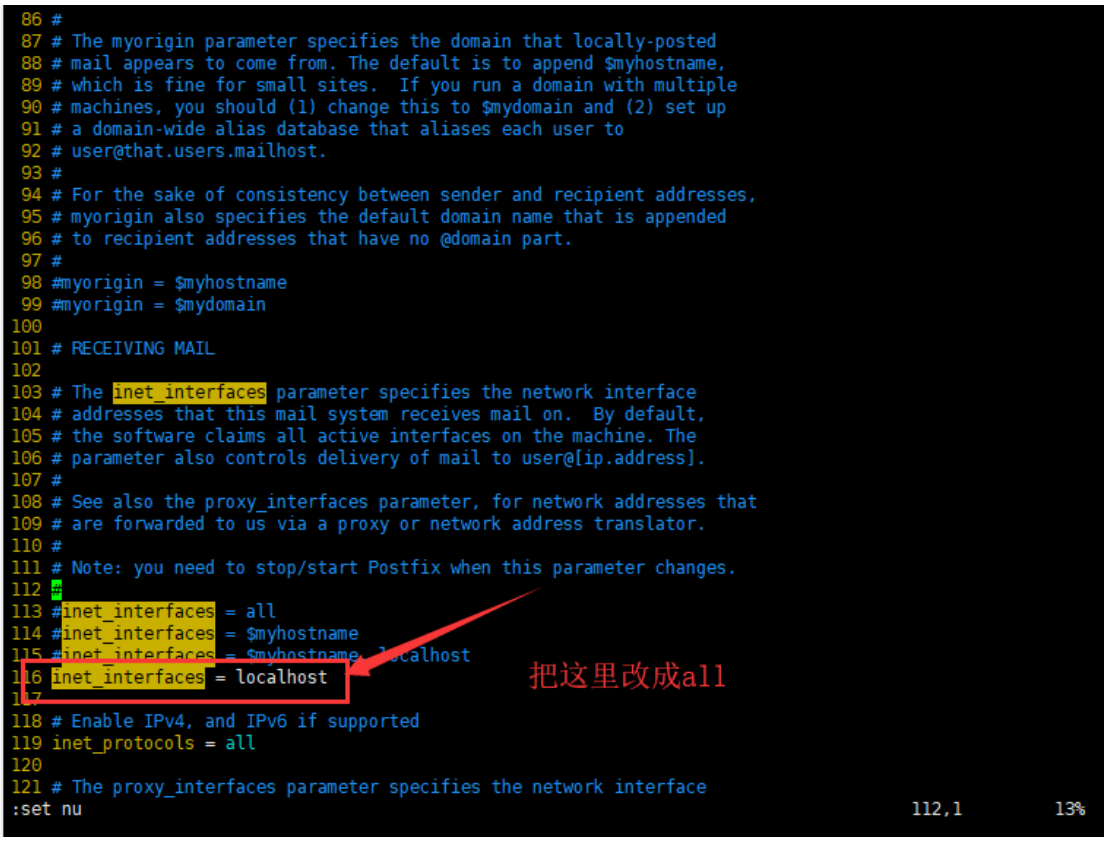
sudo yum install postfix

设置 Postfix 开机自启动。

sudo systemctl enable postfix

启动 Postfix 服务。

输入命令 vim /etc/postfix/main.cf 打开 main.cf 文件并找到下图内容：



将这行代码改为 inet_interfaces = all，然后按 Esc 键，然后输入:wq 并回车以保存并关闭 main.cf 文件。

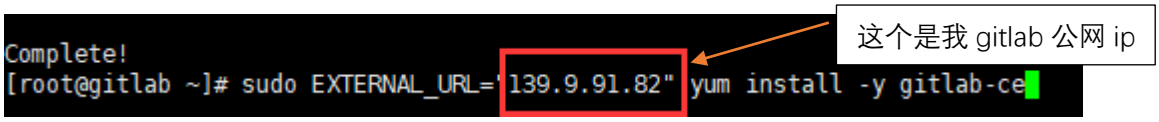
输入命令 sudo systemctl start postfix 启动 Postfix 服务。

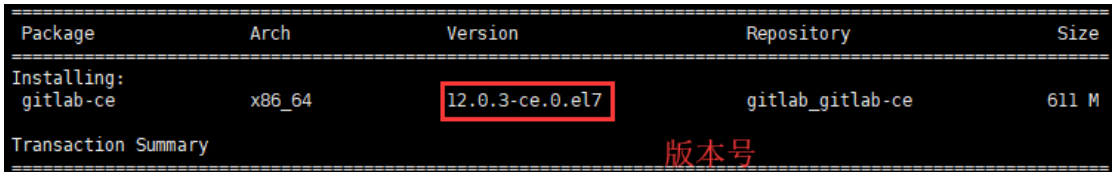
添加 GitLab 软件包仓库。

curl https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.rpm.sh | sudo bash

安装 GitLab。

sudo EXTERNAL_URL="GitLab 服务器的公网 IP 地址" yum install -y gitlab-ce





安装完成会出现如下图片：

说明已经自动启动 getlab 服务，如没有出现以上情况则说明硬件配置太低或输入 systemctl restart gitlab-runsvdir.service 重启 gitlab 服务

其他操作命令：

sudo gitlab-ctl status	查看服务的状态
sudo gitlab-ctl start	启动
sudo gitlab-ctl stop	关闭
sudo gitlab-ctl restart	重启

[root@gitlab ~]# vim /etc/gitlab/gitlab.rb #gitlab 配置文件

```
# GitLab configuration settings
##! This file is generated during initial installation and **is not** modified
##! during upgrades.
##! Check out the latest version of this file to know about the different
##! settings that can be configured by this file, which may be found at:
##! https://gitlab.com/gitlab-org/omnibus-gitlab/raw/master/files/gitlab-config-template/gitlab.rb.template

## GitLab URL
##! URL on which GitLab will be reachable.
##! For more details on configuring external_url see:
##! https://docs.gitlab.com/omnibus/settings/configuration.html#configuring-the-external-url-for-gitlab
external_url 'http://139.9.91.82'

## Roles for multi-instance GitLab
##! The default is to have no roles enabled, which results in GitLab running as an all-in-one instance.
##! Options:
##!   redis_sentinel_role redis_master_role redis_slave_role geo_primary_role geo_secondary_role
##! For more details on each role, see:
##! https://docs.gitlab.com/omnibus/roles/README.html#roles
##!
# roles ['redis_sentinel_role', 'redis_master_role']

## Legend
##! The following notations at the beginning of each line may be used to
##! differentiate between components of this file and to easily select them using
##! a regex.
##! ## Titles, subtitles etc
##! ##! More information - Description, Docs, Links, Issues etc.
##! Configuration settings have a single # followed by a single space at the
##! beginning; Remove them to enable the setting.

##! **Configuration settings below are optional.**
:set nu                                                    1,1           Top
```

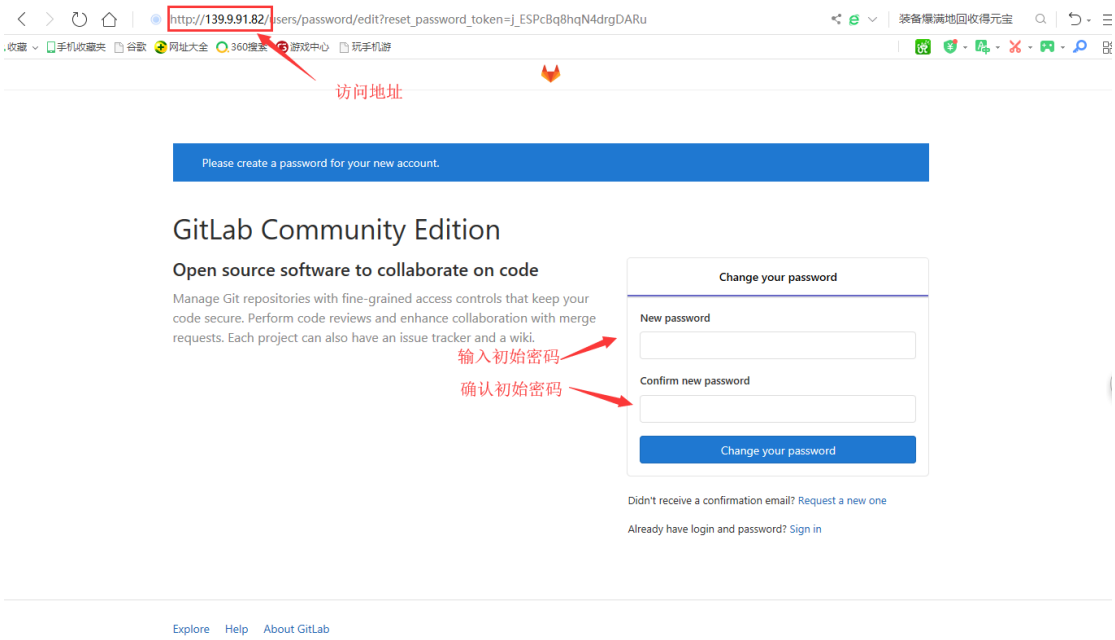
这里可以修改访问端口

如果修改了配置文件，需要执行 gitlab-ctl reconfigure 进行更新配置（只要修改了该文件，必须执行该命令才能生效）

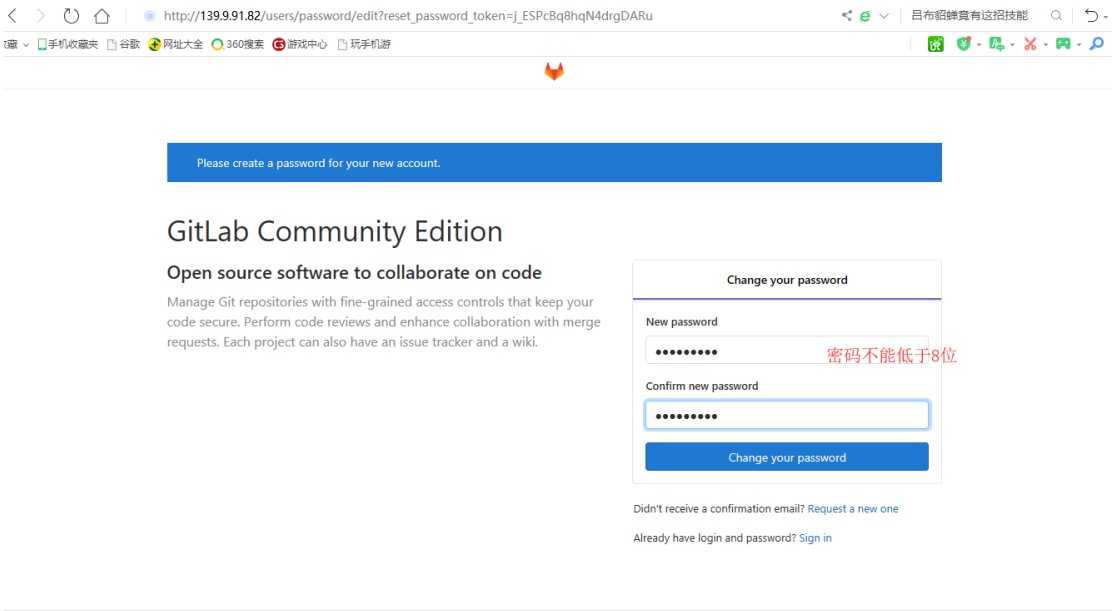
执行 gitlab-ctl restart 重启服务，我这里不做修改

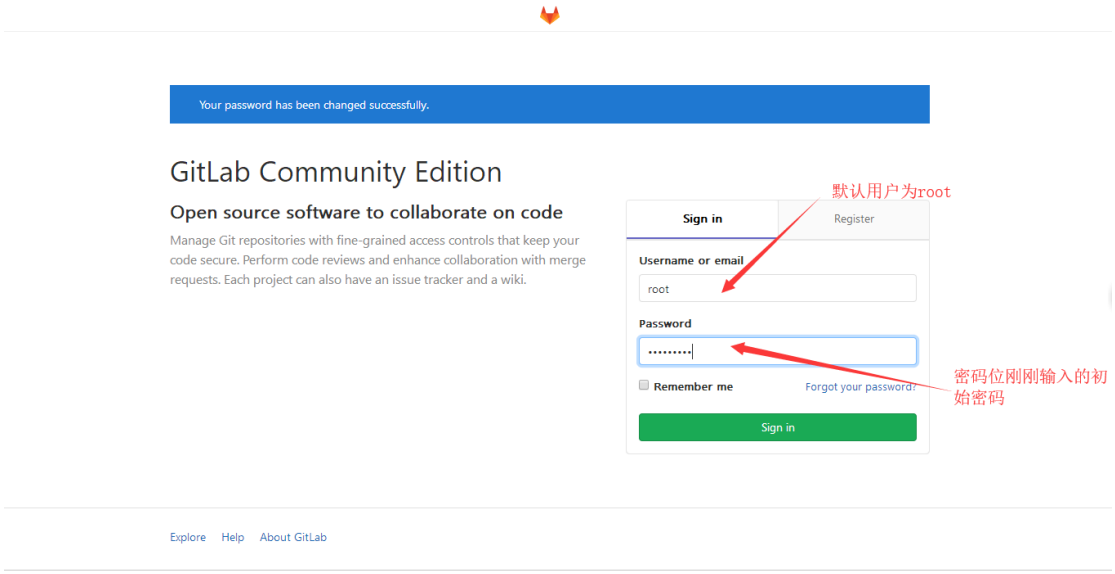
服务启动成功后即可通过公网 IP 去访问，如果修改了端口，IP 后面就要加上端口去访问

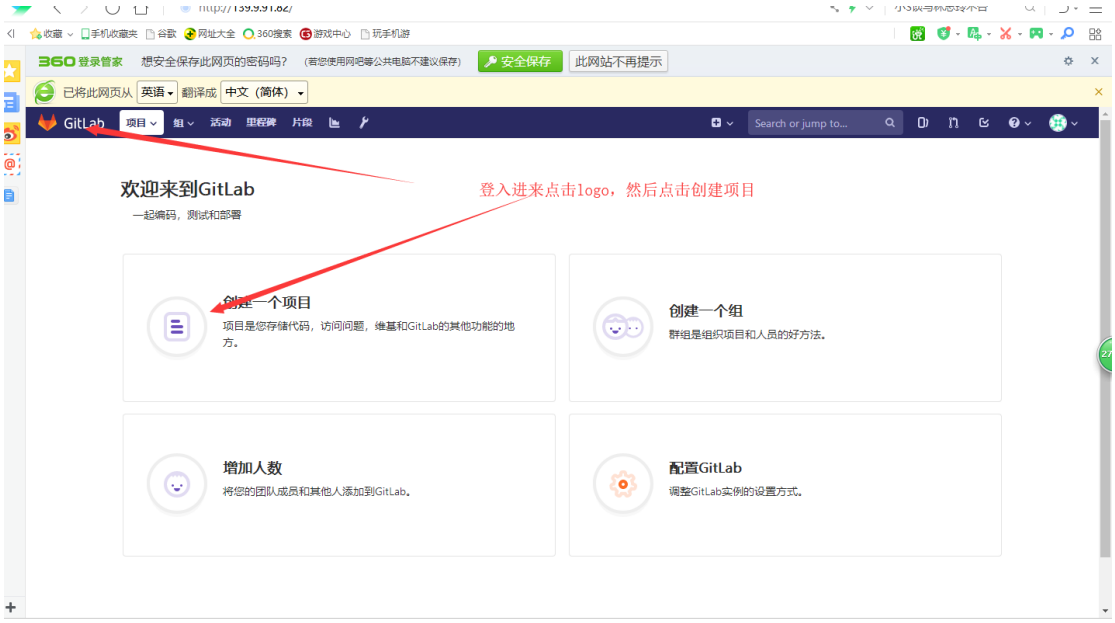
使用浏览器访问 GitLab 服务器的公网 IP 地址，显示如下页面，说明环境搭建成功。

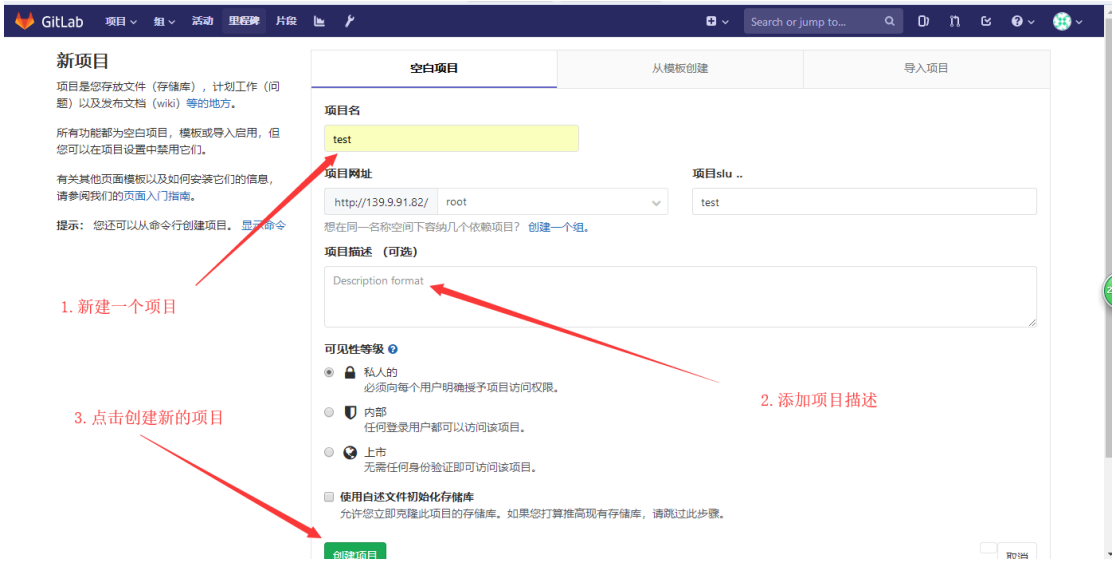


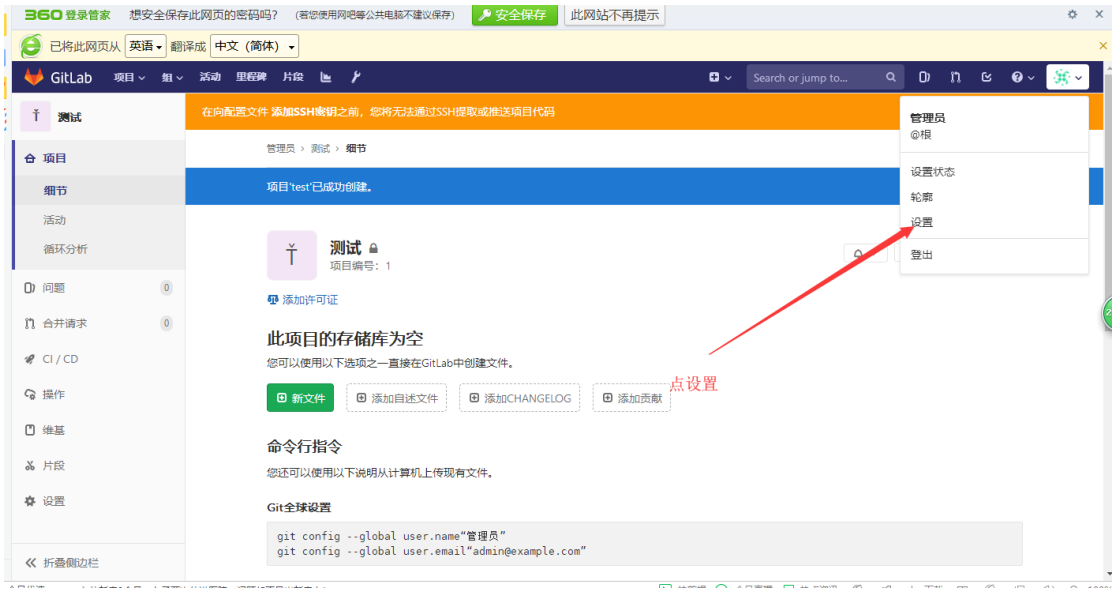
管理员默认为 root 用户，初始密码不能低于 8 位



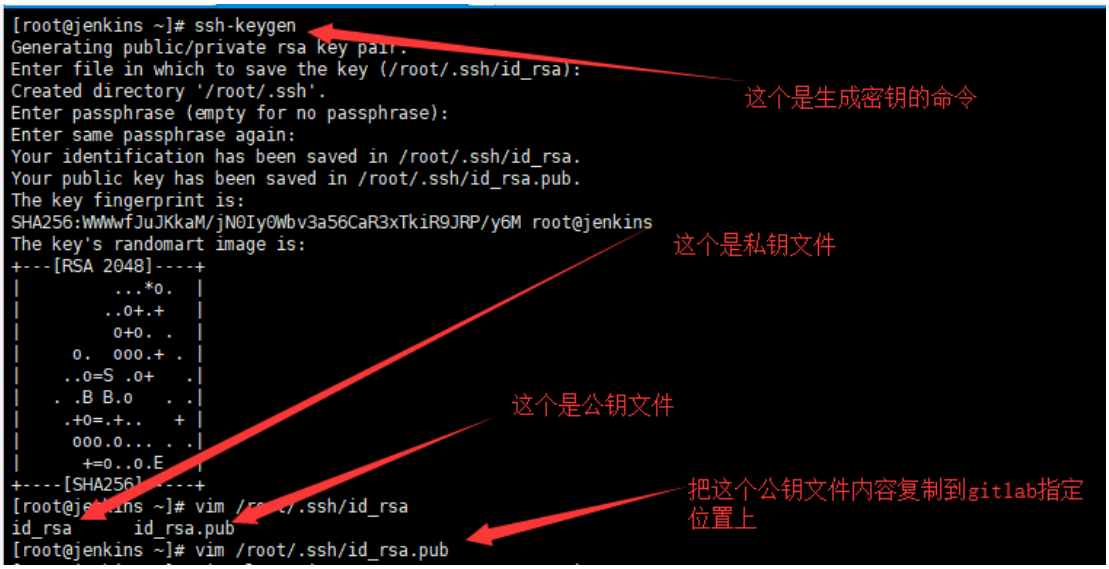



登入进来点击logo，然后点击创建项目
欢迎来到GitLab
一起编码，测试和部署
创建一个项目
项目是您存储代码，访问问题，维基和GitLab的其他功能的地方。
创建一个组
群组是组织项目和人员的好方法。
增加人数
将您的团队成员和其他人添加到GitLab。
配置GitLab
调整GitLab实例的设置方式。

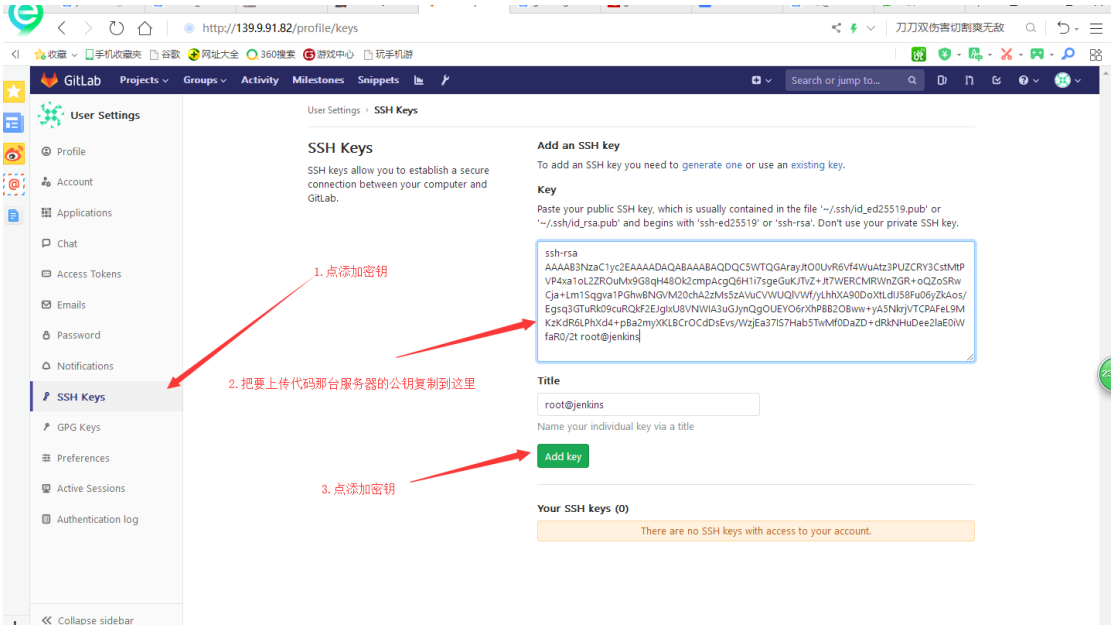

新项目
空白项目
从模板创建
导入项目
项目名
test
项目网址
http://139.9.91.82/
root
项目slu ..
test
项目描述 （可选）
Description format
可见性等级
私人的
必须向每个用户明确授予项目访问权限。
内部
任何登录用户都可以访问该项目。
上市
无需任何身份验证即可访问该项目。
使用自述文件初始化存储库
允许您立即克隆此项目的存储库。如果您打算推高现有存储库，请跳过此步骤。
创建项目
1. 新建一个项目
2. 添加项目描述
3. 点击创建新的项目


在向配置文件 添加SSH密钥 之前，您将无法通过SSH提取或推送项目代码
项目'test'已成功创建。
测试
项目编号：1
添加许可证
此项目的存储库为空
您可以使用以下选项之一直接在GitLab中创建文件。
新文件
添加自述文件
添加CHANGELOG
添加贡献
命令行指令
您还可以使用以下说明从计算机上传现有文件。
Git全球设置
git config --global user.name"管理员"
git config --global user.email"admin@example.com"
管理员
@根
设置状态
轮廓
设置
登出
点设置

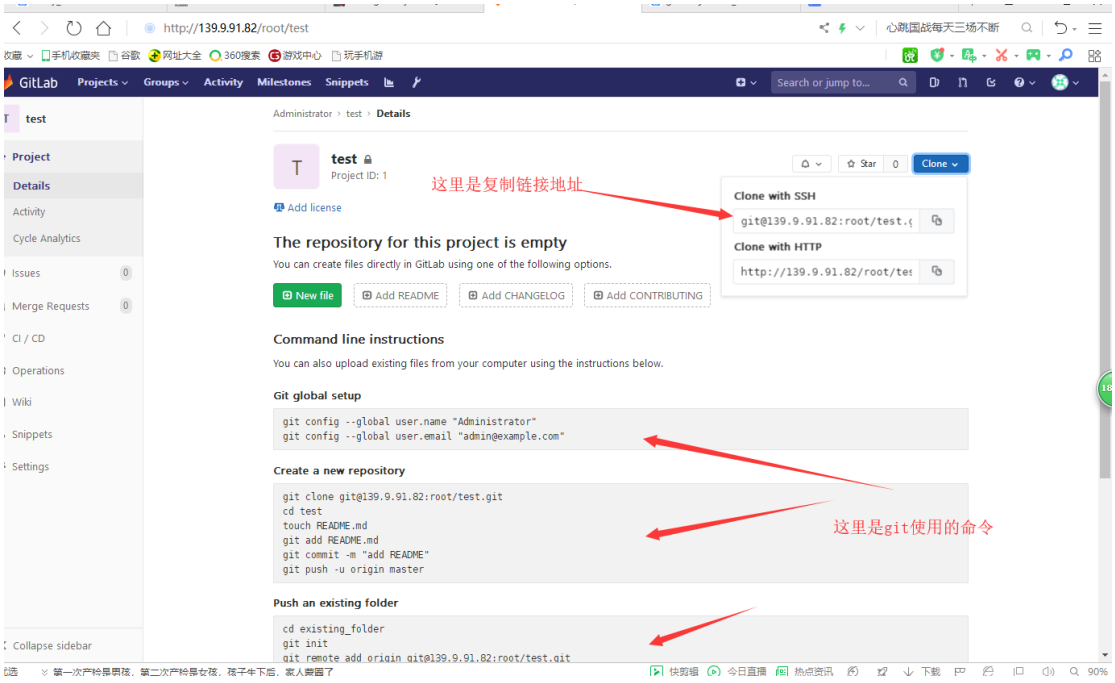
把要克隆或要上传代码这台服务器的公钥 cat /root/.ssh/id_rsa.pub 打开公钥文件复制到gitlab 服务器指定位置上



添加公钥



#克隆代码仓库，然后测试是否能够上传代码到 gitlab



yum -y install git　　#安装 git

git clone git@139.9.91.82:root/test.git　　#克隆远程仓库

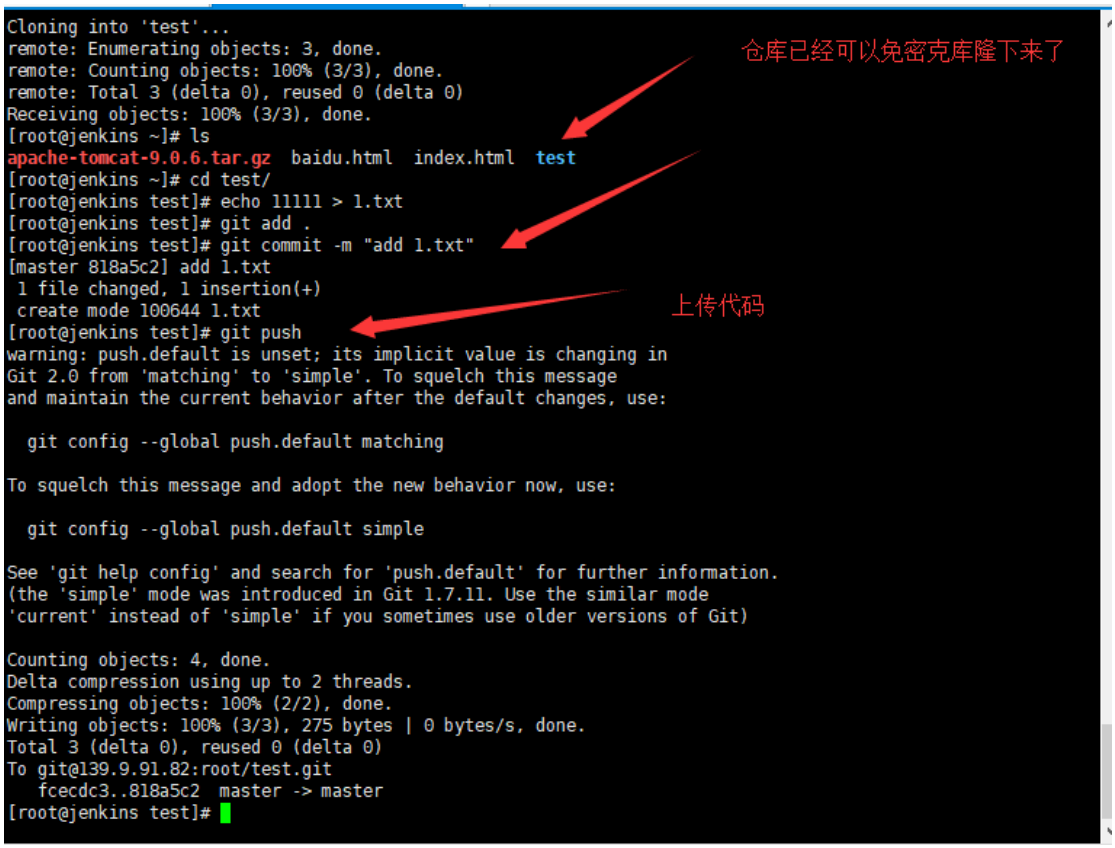
cd test/

echo 11111 > 1.txt

git add .

git commit -m "add 1.txt"

git push



配置 Jenkins

系统要求

最低推荐配置:

256MB 可用内存

1GB 可用磁盘空间(作为一个 Docker 容器运行 jenkins 的话推荐 10GB)为小团队推荐的

硬件配置:

1GB+可用内存

50 GB+ 可用磁盘空间

软件配置:

Java 8—无论是 Java 运行时环境（JRE）还是 Java 开发工具包（JDK）都可以。

安装 JDK

yum install -y java-1.8.0-openjdk java-1.8.0-openjdk-devel

安装 tomcat

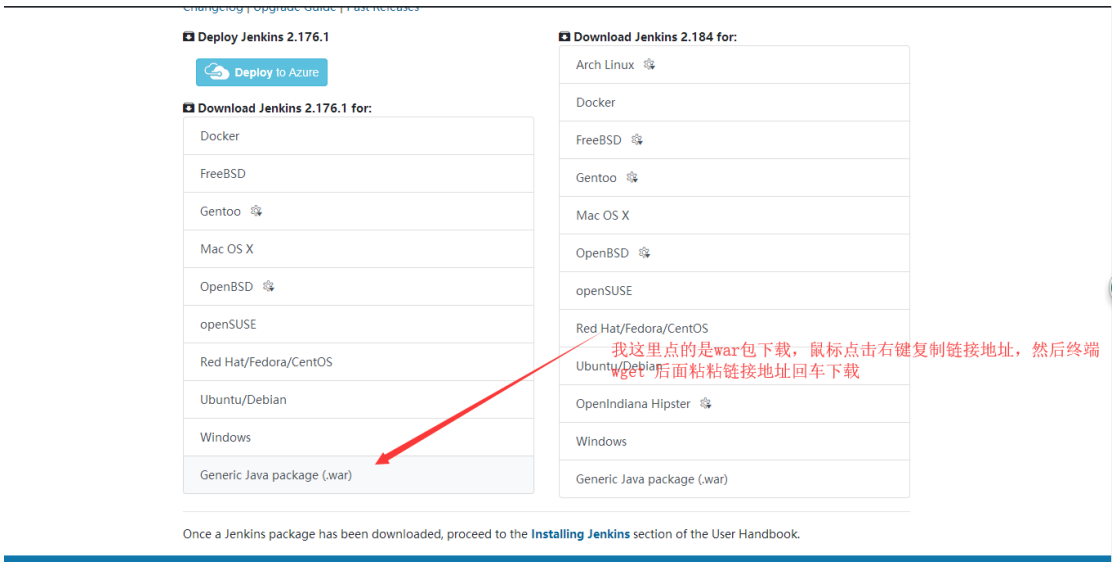
[root@jenkins ~]# tar -xf apache-tomcat-9.0.6.tar.gz

[root@jenkins ~]# mv apache-tomcat-9.0.6 /usr/local/tomcat

```
[root@jenkins ~]# tar -xf apache-tomcat-9.0.6.tar.gz
[root@jenkins ~]# ls
apache-tomcat-9.0.6  apache-tomcat-9.0.6.tar.gz  baidu.html  index.html
[root@jenkins ~]# mv apache-tomcat-9.0.6 /usr/local/tomcat
[root@jenkins ~]#
```

访问 jenkins 官网 https://jenkins.io/zh/ 下载 Jenkins

Jenkins 下载



wget http://mirrors.jenkins.io/war-stable/latest/jenkins.war

两种方式使用其中一种就可以下载了

Jenkins的RedHat Linux RPM包

要使用此存储库，请运行以下命令：

也可以生成jenkins仓库

```
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
```

如果您以前从Jenkins导入了密钥，那么“rpm --import”将失败，因为您已经有了密钥。请忽略它并继续前进。

您将需要显式安装Java运行时环境，因为Oracle的Java RPM不正确并且无法注册为提供Java依赖项。因此，在Java上添加显式依赖性要求会强制安装OpenJDK JVM。

- 2.164（2019-02）和更新版本：Java 8或Java 11
- 2.54（2017-04）和更新版本：Java 8
- 1.612（2015-05）及更新版本：Java 7

有了这个设置，Jenkins包可以安装：

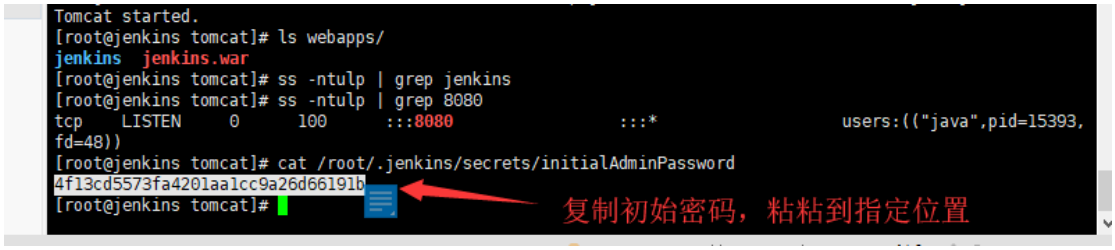
```
yum安装jenkins
```

[root@jenkins ~]# ls
apache-tomcat-9.0.6.tar.gz baidu.html index.html jenkins.war
[root@jenkins ~]# cd /usr/local/tomcat/ #切换到 tomcat 目录
[root@jenkins tomcat]# pwd
/usr/local/tomcat
[root@jenkins tomcat]# rm -rf webapps/* #删除 tomcat 根目录下其他默认文件
[root@jenkins tomcat]# ls webapps/
[root@jenkins tomcat]# mv /root/jenkins.war /usr/local/tomcat/webapps/ #移动下载好

的 jenkins.war 包到 tomcat 网页根目录下

```
[root@jenkins tomcat]# ls
bin  conf  lib  LICENSE  logs  NOTICE  RELEASE-NOTES  RUNNING.txt  temp  webapps  work
[root@jenkins tomcat]# ./bin/startup.sh          #启动 tomcat 服务
Using CATALINA_BASE:   /usr/local/tomcat
Using CATALINA_HOME:   /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
Tomcat started.
[root@jenkins tomcat]# ls webapps/          #tomcat 服务启动会自动解 jenkins.war 包
jenkins  jenkins.war
```

然后访问 http://139.159.189.191:8080/jenkins 见到如下图片就说明安装成功：



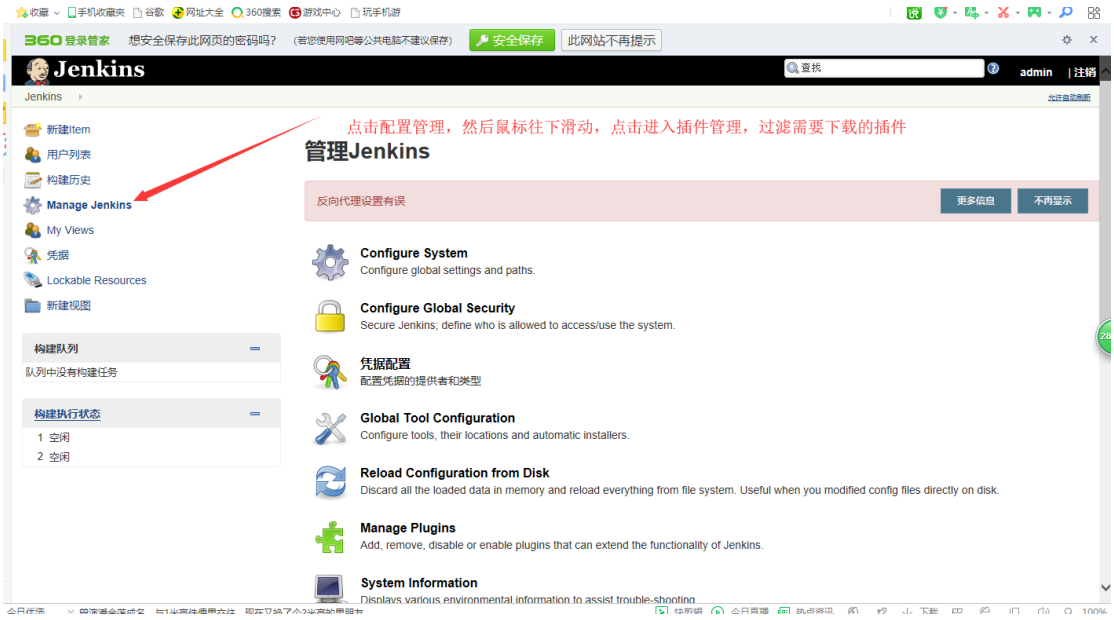
插件安装：

打×的插件是由于网络传输导致的安装失败，后面再重新安装即可。
插件安装可以自定义，可以用社区推荐安装，工作需要的插件一定要安装上

安装完成出现如下页面：


http://139.159.189.191:8080/jenkins/
登录管家
想安全保存此网页的密码吗？
（若您使用网吧等公共电脑不建议保存）
安全保存
此网站不再提示
新手入门
实例配置
Jenkins URL:
http://139.159.189.191:8080/jenkins/
Jenkins URL 用于给各种Jenkins资源提供绝对路径链接的根地址。这意味着对于很多Jenkins特色是需要正确设置的，例如：邮件通知、PR状态更新以及提供给构建步骤的BUILD_URL环境变量。
推荐的默认值显示在首次保存，如果可能的话这是根据当前请求生成的。最佳实践是要设置这个值，用户可能会需要用到。这将会避免在分享或者查看链接时的困惑。
Jenkins 2.176.1
现在不要
保存并完成

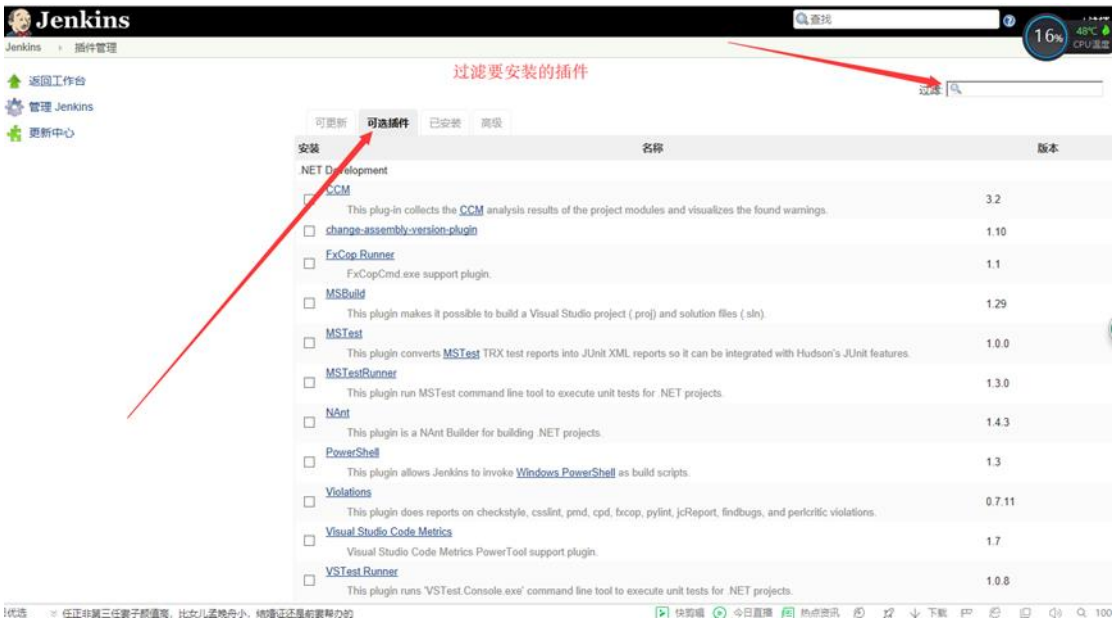

360 登录管家
想安全保存此网页的密码吗？
（若您使用网吧等公共电脑不建议保存）
安全保存
此网站不再提示
新手入门
Jenkins已就绪！
Jenkins安装已完成。
开始使用Jenkins
点击进入Jenkins界面
Jenkins 2.176.1


Jenkins
admin | 注销
Jenkins
新建Item
用户列表
构建历史
Manage Jenkins
My Views
凭据
Lockable Resources
新建视图
构建队列
队列中没有构建任务
构建执行状态
1 空闲
2 空闲
点击配置管理，然后鼠标往下滑动，点击进入插件管理，过滤需要下载的插件
管理Jenkins
反向代理设置有误
更多信息
不再显示
Configure System
Configure global settings and paths.
Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.
凭据配置
配置凭据的提供者和类型
Global Tool Configuration
Configure tools, their locations and automatic installers.
Reload Configuration from Disk
Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.
Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
System Information

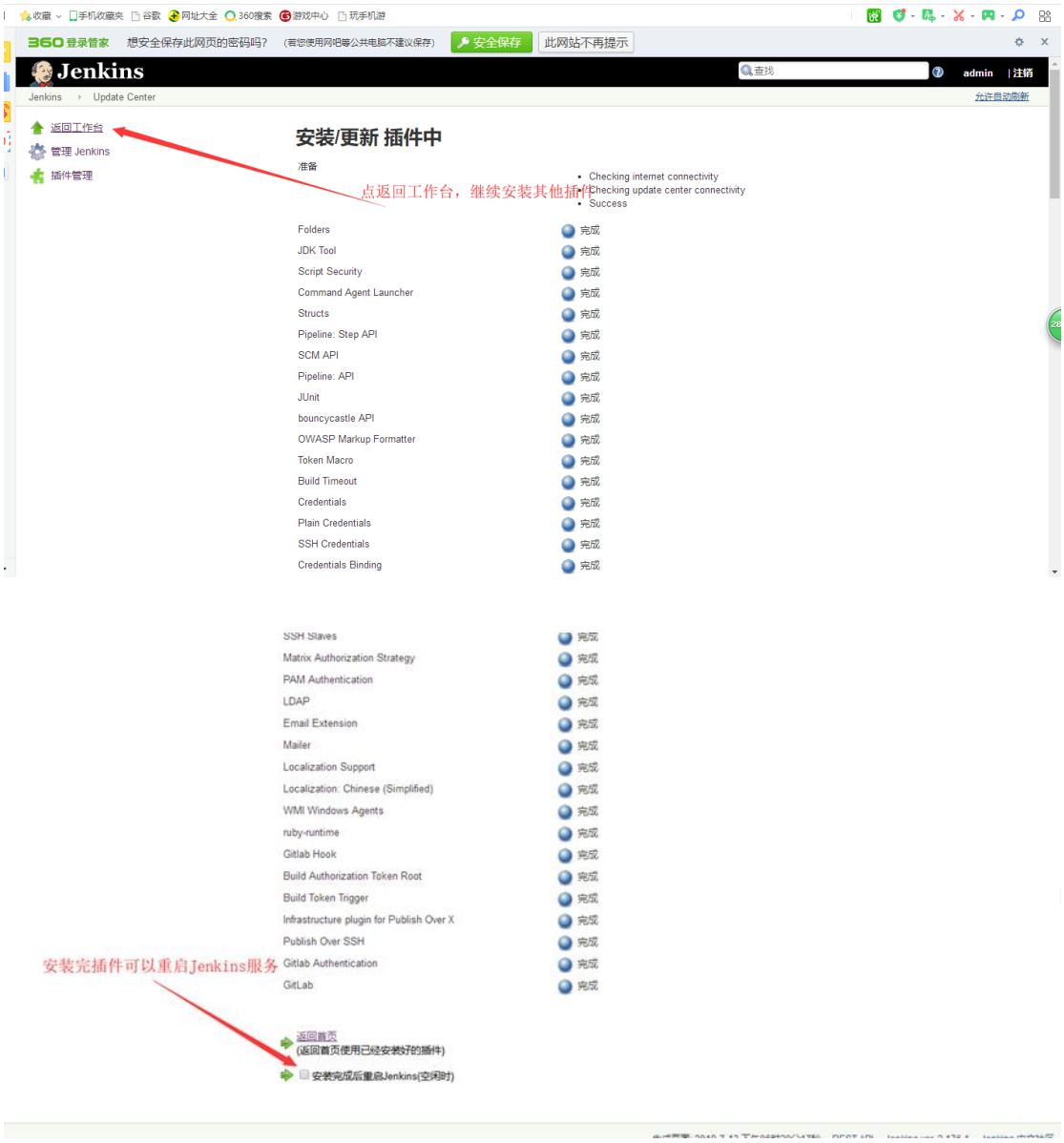

Jenkins
admin | 注销
詹金斯
允许自动刷新
新建项目
用户列表
构建历史
管理詹金斯
我的观点
凭据
可锁定的资源
新建视图
构建队列
队列中没有构建任务
构建执行状态
1 空闲
2 空闲
管理詹金斯
反向代理设置有误
更多信息
不再显示
配置系统
配置全局设置和路径。
配置全局安全性
安全詹金斯，定义允许谁访问/使用系统。
凭据配置
配置凭据的提供者和类型
全局工具配置
配置工具，它们的位置和自动安装程序。
从磁盘重新加载配置
丢弃内存中的所有已加载数据，并从文件系统重新加载所有内容。直接在磁盘上修改配置文件时很有用。
管理插件
添加，删除，禁用或启用可以扩展Jenkins功能的插件。
点击插件管理，下载需要的插件

如果插件没安装完会出现如下页面，点纠正安装相关插件就好了：

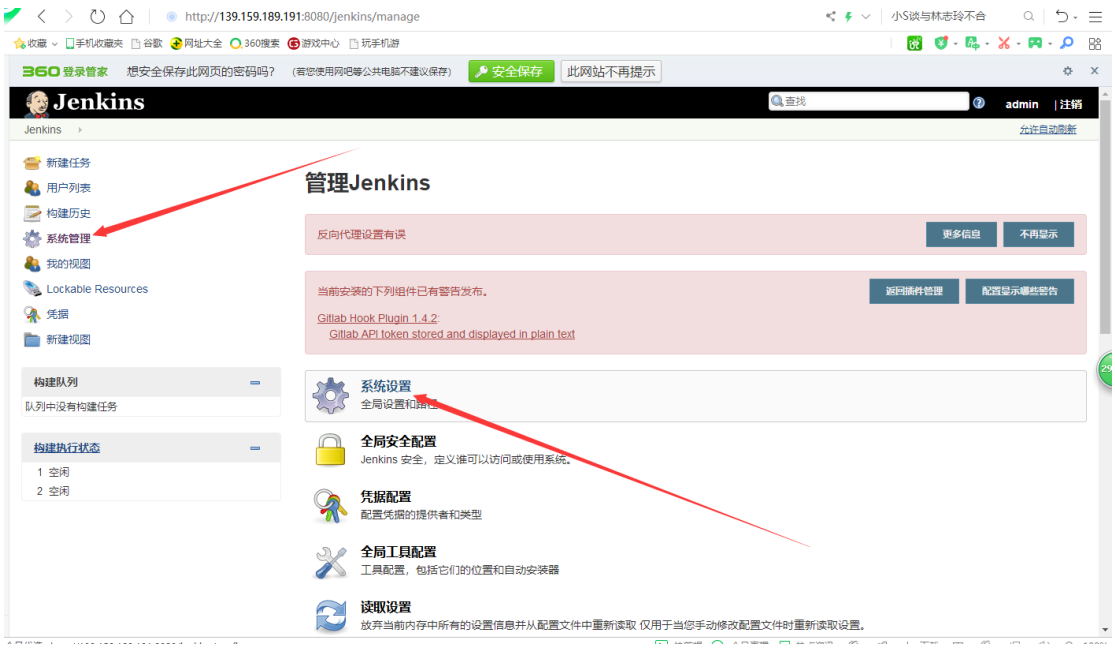
必须需要添加的 6 个插件 : Gitlab Hook、Build Authorization Token Root、Publish Over SSH、Gitlab Authentication、Gitlab、Git Parameter



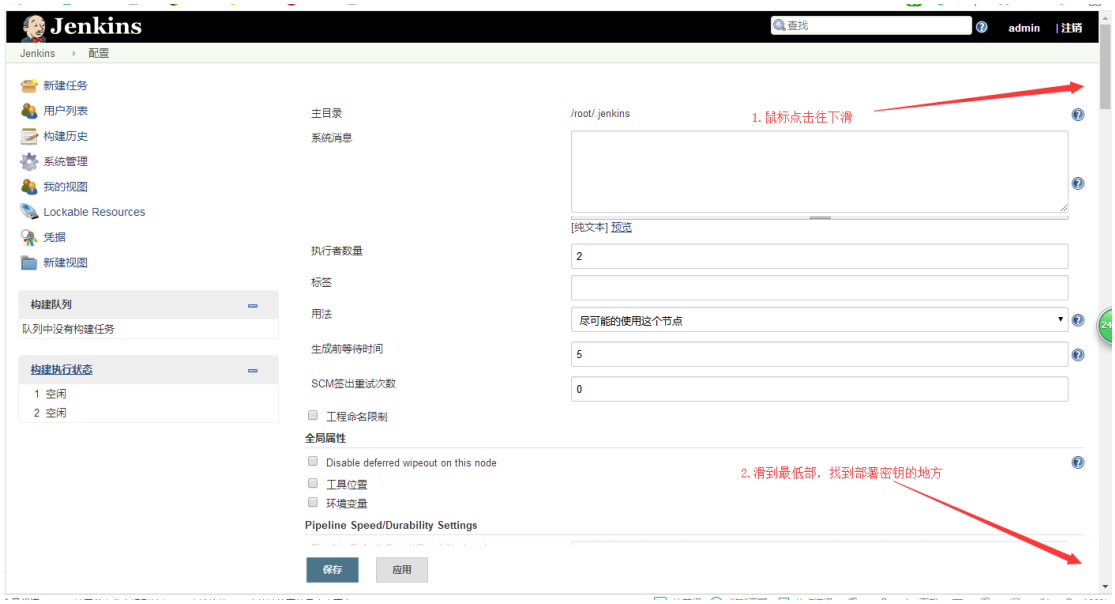

Jenkins
插件管理
返回工作台
管理 Jenkins
更新中心
1. 点可选插件
2. 搜索过滤需要安装的插件
3. 然后勾选
4. 然后点安装
过滤:
Gitlab Hook
可更新
可选插件
已安装
高级
安装
名称
版本
Gitlab Hook
Enables Gitlab web hooks to be used to trigger SMC polling on Gitlab projects
Warning: This plugin version may not be safe to use. Please review the following security notices:
1.4.2
Gitlab API token stored and displayed in plain text
直接安装
下载待重启后安装
36 分 之前获取了更新信息
立即获取
生成页面: 2019-7-13 下午06时06分21秒
REST API
Jenkins ver. 2.176.1
Jenkins 中文社区


360登录管家
想安全保存此网页的密码吗?
(若您使用网吧等公共电脑不建议保存)
安全保存
此网站不再提示
Jenkins
admin
注销
Jenkins
Update Center
允许自动刷新
返回工作台
管理 Jenkins
插件管理
安装/更新 插件中
准备
Checking internet connectivity
Checking update center connectivity
Success
点返回工作台，继续安装其他插件
Folders 完成
JDK Tool 完成
Script Security 完成
Command Agent Launcher 完成
Structs 完成
Pipeline: Step API 完成
SCM API 完成
Pipeline: API 完成
JUnit 完成
bouncycastle API 完成
OWASP Markup Formatter 完成
Token Macro 完成
Build Timeout 完成
Credentials 完成
Plain Credentials 完成
SSH Credentials 完成
Credentials Binding 完成
SSH Slaves 完成
Matrix Authorization Strategy 完成
PAM Authentication 完成
LDAP 完成
Email Extension 完成
Mailer 完成
Localization Support 完成
Localization: Chinese (Simplified) 完成
WMI Windows Agents 完成
ruby-runtime 完成
Gitlab Hook 完成
Build Authorization Token Root 完成
Build Token Trigger 完成
Infrastructure plugin for Publish Over X 完成
Publish Over SSH 完成
Gitlab Authentication 完成
GitLab 完成
安装完插件可以重启Jenkins服务
返回首页
(返回首页使用已经安装好的插件)
安装完成后重启Jenkins(空闲时)

配置 jenkins

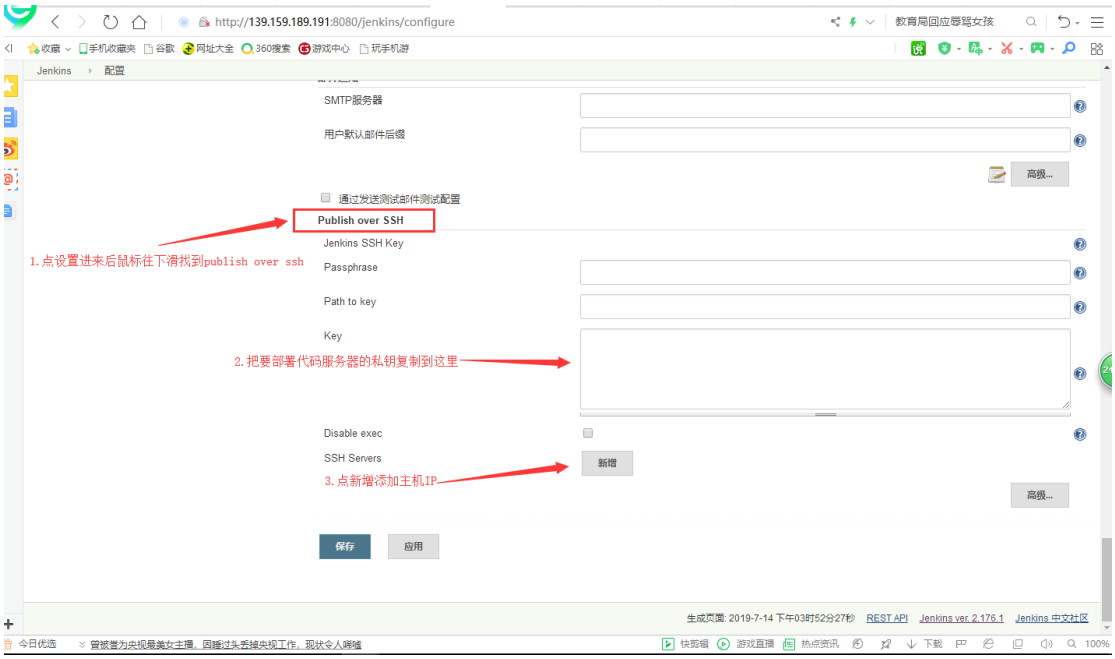


进来系统设置如下图，然后鼠标往下滑：

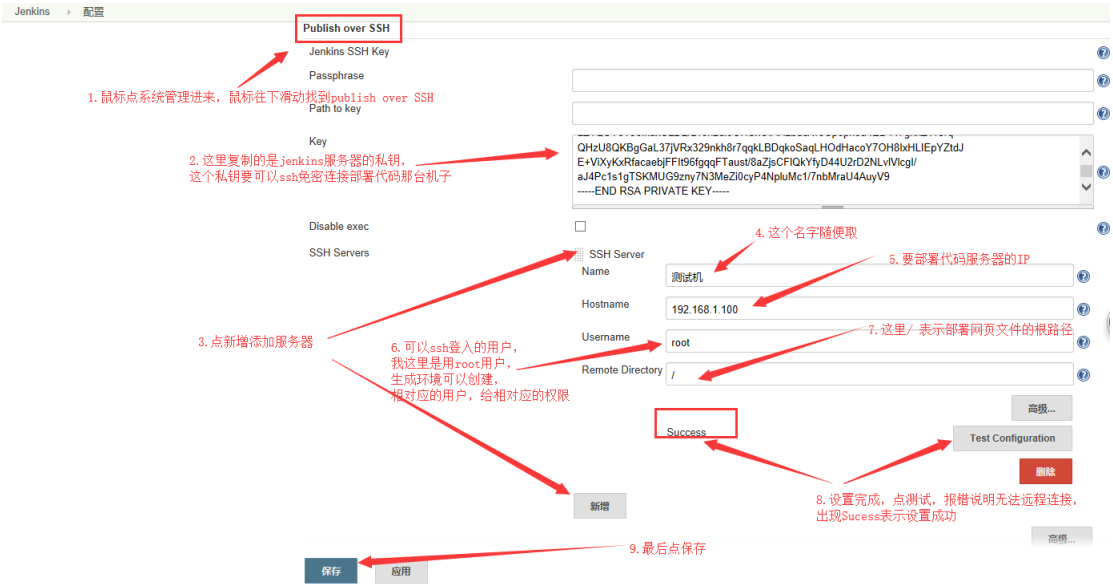


部署密钥：

#添加主机：系统管理 - 系统设置 找到 Publish over SSH



开始添加要部署代码的主机，注意一定是要能够 ssh 登陆的用户。



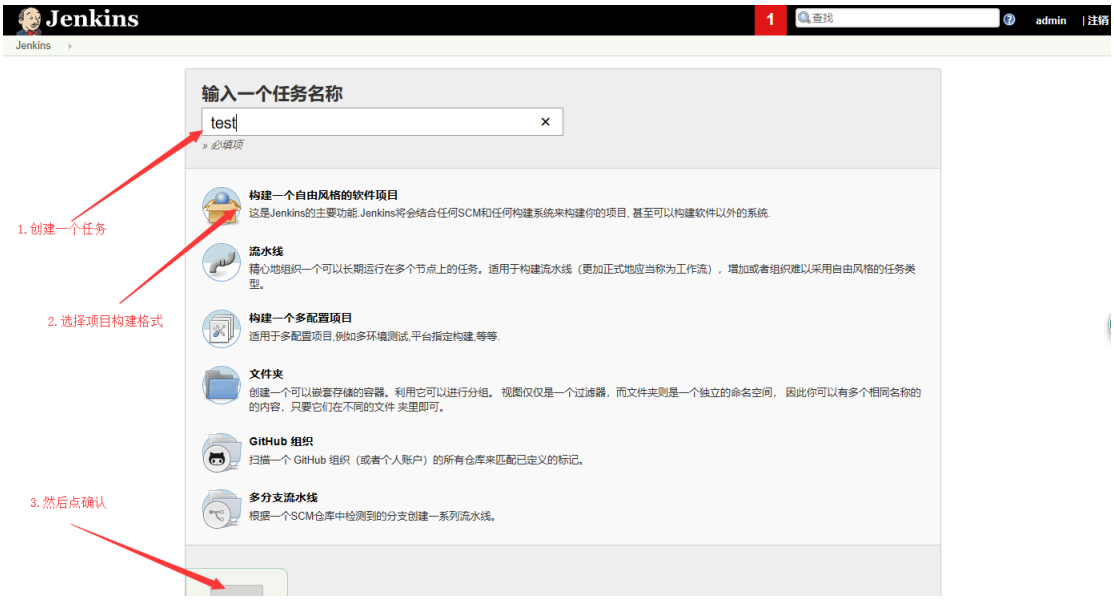
这里要注意的是要先把 jenkins 服务器的公钥传送给要部署代码的那台机然后再点测试

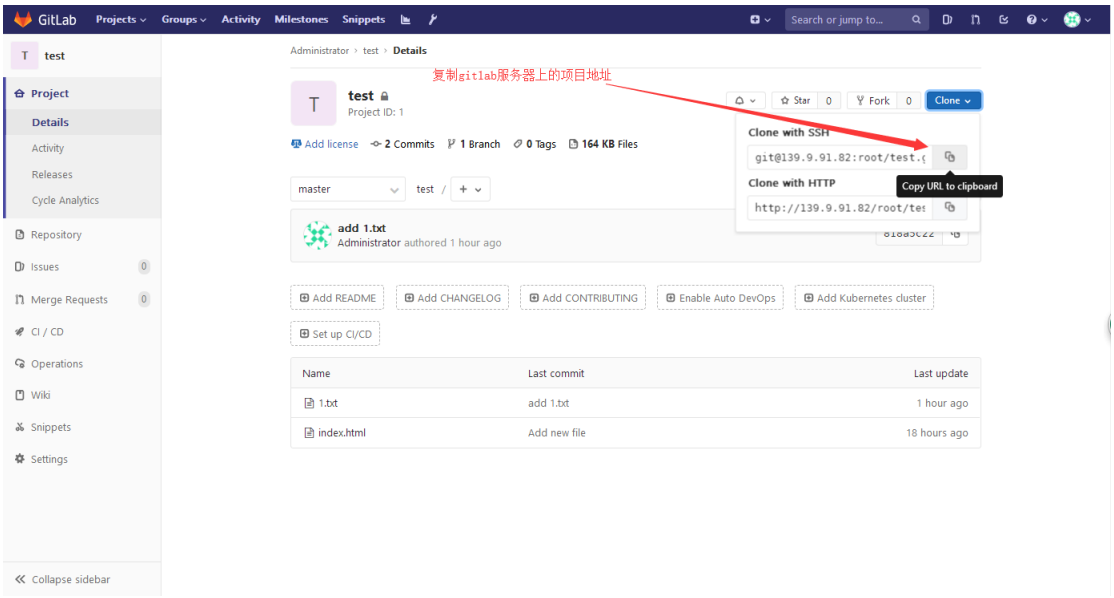
```
[root@jenkins ~]# ssh-copy-id 192.168.1.100
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '192.168.1.100 (192.168.1.100)' can't be established.
ECDSA key fingerprint is SHA256:sqx8Ejs+BMrzX038mG7tl8edXeoU+qoaNTa6236hIQU.
ECDSA key fingerprint is MD5:d3:25:81:05:79:d5:38:45:88:6e:94:4f:78:7a:44:d7.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already i
nstalled
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install th
e new keys
root@192.168.1.100's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh '192.168.1.100'"
and check to make sure that only the key(s) you wanted were added.

[root@jenkins ~]# cat /root/.ssh/id_rsa
```

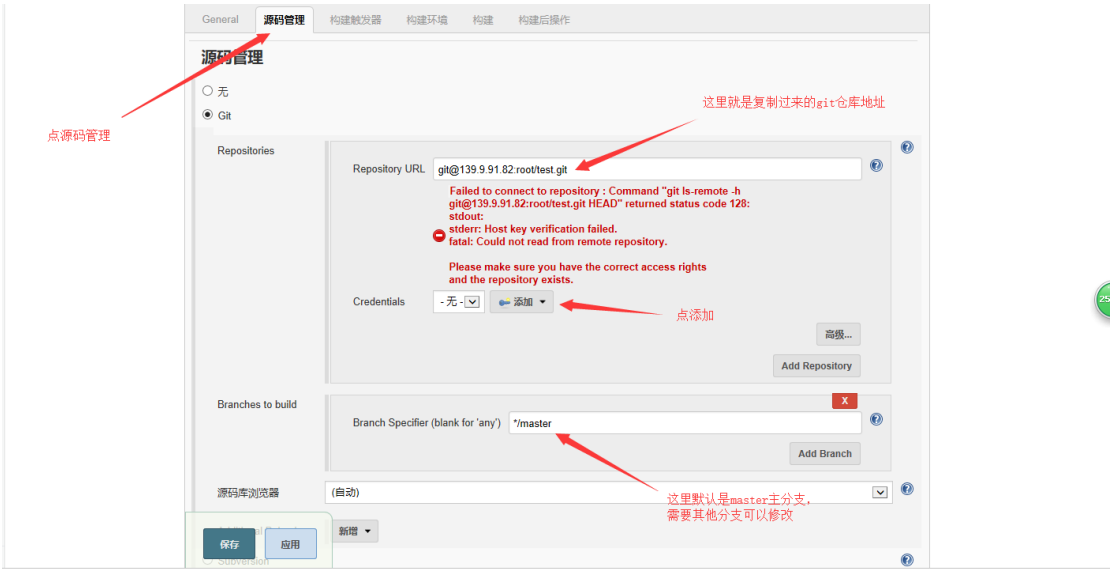
拷贝Jenkins服务器的公钥给要部署代码的那台机器

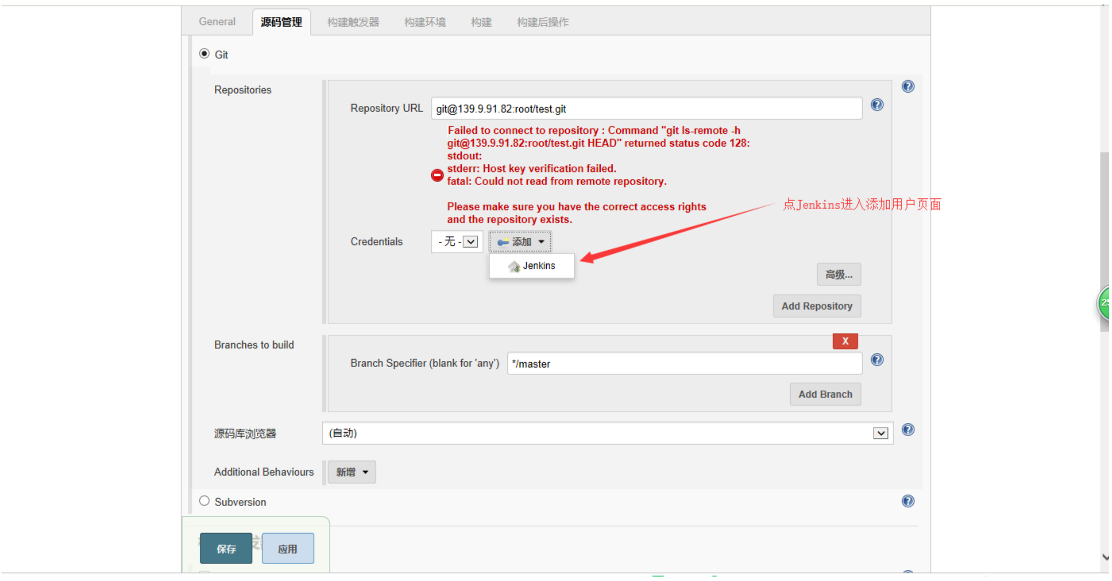
输入要部署代码那台机子的登入密码

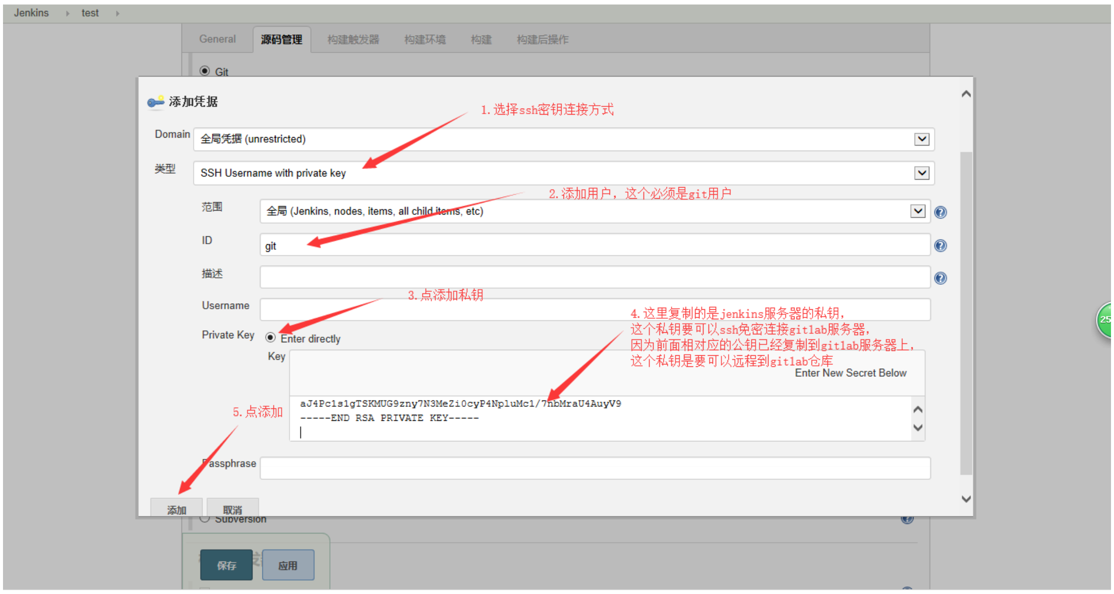
#添加完主机之后，新建一个项目，开始配置构建操作

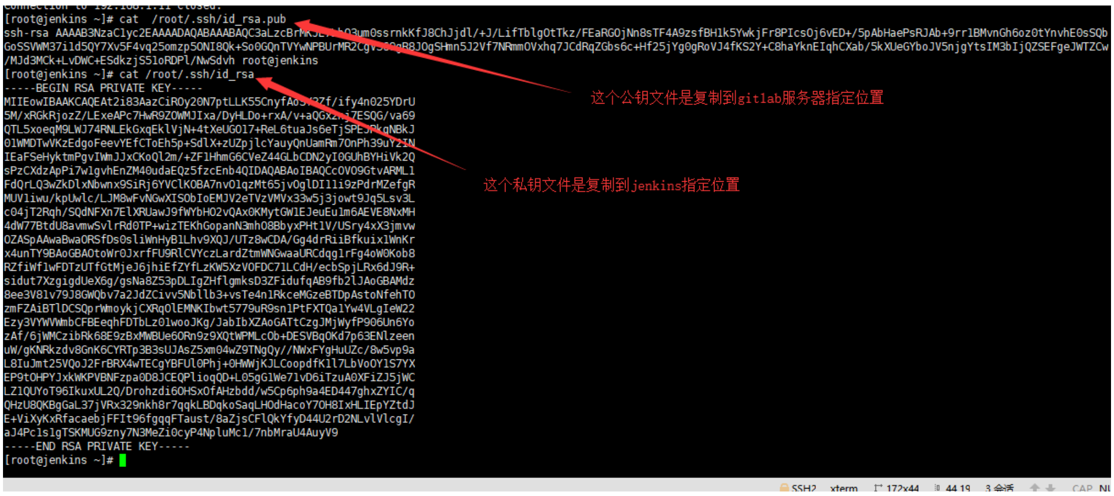

Jenkins
admin | 注销
Jenkins
输入一个任务名称
test
» 必填项
构建一个自由风格的软件项目
这是Jenkins的主要功能.Jenkins将会结合任何SCM和任何构建系统来构建你的项目, 甚至可以构建软件以外的系统.
流水线
精心地组织一个可以长期运行在多个节点上的任务。适用于构建流水线（更加正式地应当称为工作流），增加或者组织难以采用自由风格的任务类型。
构建一个多配置项目
适用于多配置项目,例如多环境测试,平台指定构建,等等.
文件夹
创建一个可以嵌套存储的容器。利用它可以进行分组。视图仅仅是一个过滤器，而文件夹则是一个独立的命名空间，因此你可以有多个相同名称的的内容，只要它们在不同的文件夹里即可。
GitHub 组织
扫描一个 GitHub 组织（或者个人账户）的所有仓库来匹配已定义的标记。
多分支流水线
根据一个SCM仓库中检测到的分支创建一系列流水线。
1. 创建一个任务
2. 选择项目构建格式
3. 然后点确认

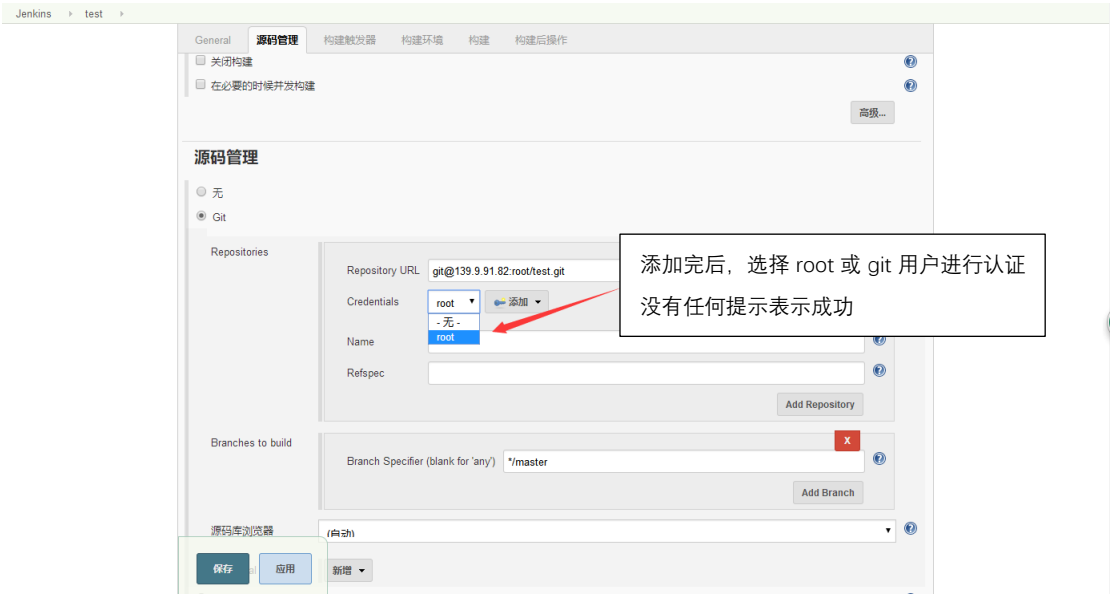

GitLab
Projects
Groups
Activity
Milestones
Snippets
Search or jump to...
test
Project
Details
Activity
Releases
Cycle Analytics
Repository
Issues
Merge Requests
CI / CD
Operations
Wiki
Snippets
Settings
Collapse sidebar
Administrator › test › Details
复制gitlab服务器上的项目地址
test
Project ID: 1
Star 0
Fork 0
Clone
Add license
2 Commits
1 Branch
0 Tags
164 KB Files
Clone with SSH
Clone with HTTP
Copy URL to clipboard
master
test
add 1.txt
Administrator authored 1 hour ago
Add README
Add CHANGELOG
Add CONTRIBUTING
Enable Auto DevOps
Add Kubernetes cluster
Set up CI/CD
Name
Last commit
Last update
1.txt
add 1.txt
1 hour ago
index.html
Add new file
18 hours ago


General
源码管理
构建触发器
构建环境
构建
构建后操作
源码管理
无
Git
点源码管理
这里就是复制过来的git仓库地址
Repositories
Repository URL
git@139.9.91.82:root/test.git
Failed to connect to repository : Command "git ls-remote -h git@139.9.91.82:root/test.git HEAD" returned status code 128:
stdout:
stderr: Host key verification failed.
fatal: Could not read from remote repository.
Please make sure you have the correct access rights and the repository exists.
Credentials
- 无 -
添加
点添加
高级...
Add Repository
Branches to build
Branch Specifier (blank for 'any')
*/master
Add Branch
这里默认是master主分支,
需要其他分支可以修改
源码库浏览器
(自动)
新增
保存
应用

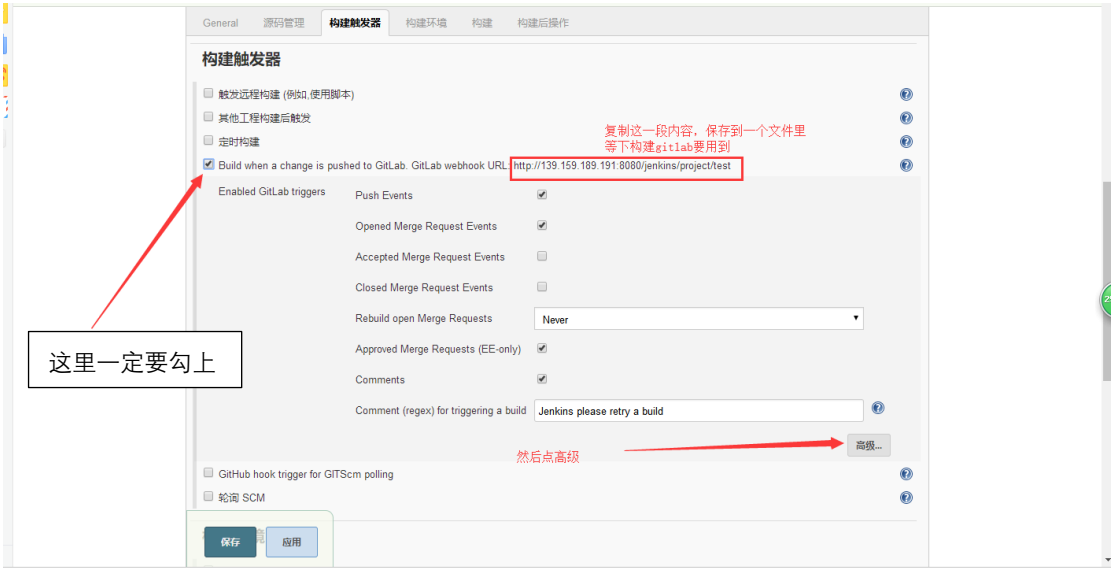

General
源码管理
构建触发器
构建环境
构建
构建后操作
Git
Repositories
Repository URL
git@139.9.91.82:root/test.git
Failed to connect to repository : Command "git ls-remote -h git@139.9.91.82:root/test.git HEAD" returned status code 128:
stdout:
stderr: Host key verification failed.
fatal: Could not read from remote repository.
Please make sure you have the correct access rights
and the repository exists.
Credentials
- 无 -
添加
Jenkins
点Jenkins进入添加用户页面
高级...
Add Repository
Branches to build
Branch Specifier (blank for 'any')
*/master
Add Branch
源码库浏览器
(自动)
Additional Behaviours
新增
Subversion
保存
应用

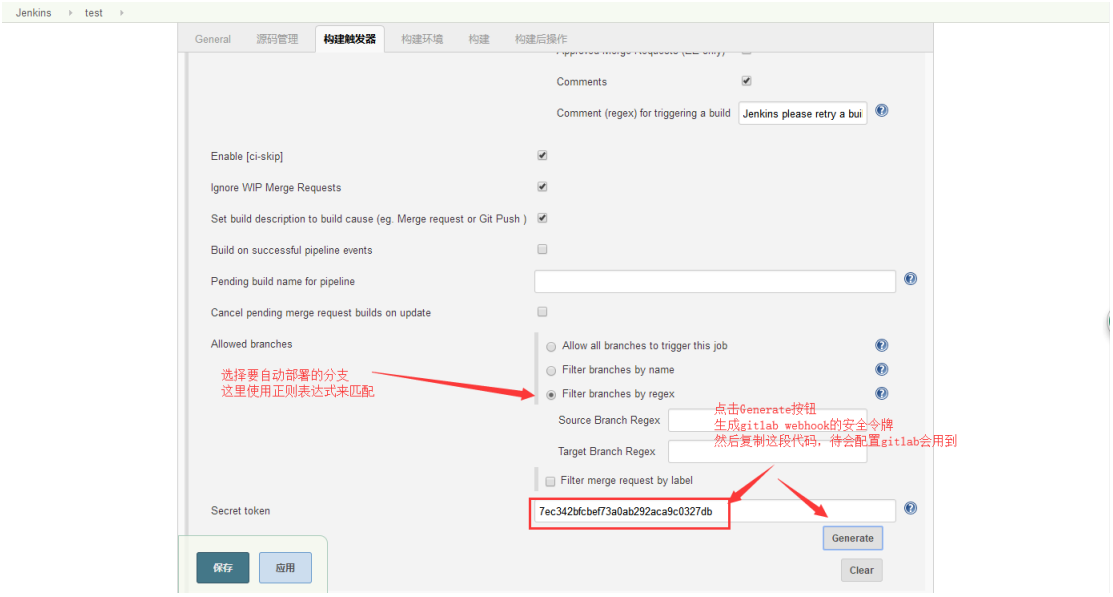

Jenkins
test
添加凭据
1. 选择ssh密钥连接方式
Domain
全局凭据 (unrestricted)
类型
SSH Username with private key
2. 添加用户，这个必须是git用户
范围
全局 (Jenkins, nodes, items, all child items, etc)
ID
git
描述
Username
3. 点添加私钥
4. 这里复制的是jenkins服务器的私钥，
这个私钥要可以ssh免密连接gitlab服务器，
因为前面相对应的公钥已经复制到gitlab服务器上，
这个私钥是要可以远程到gitlab仓库
Private Key
Enter directly
Key
Enter New Secret Below
-----END RSA PRIVATE KEY-----
5. 点添加
Passphrase
添加
取消
保存
应用

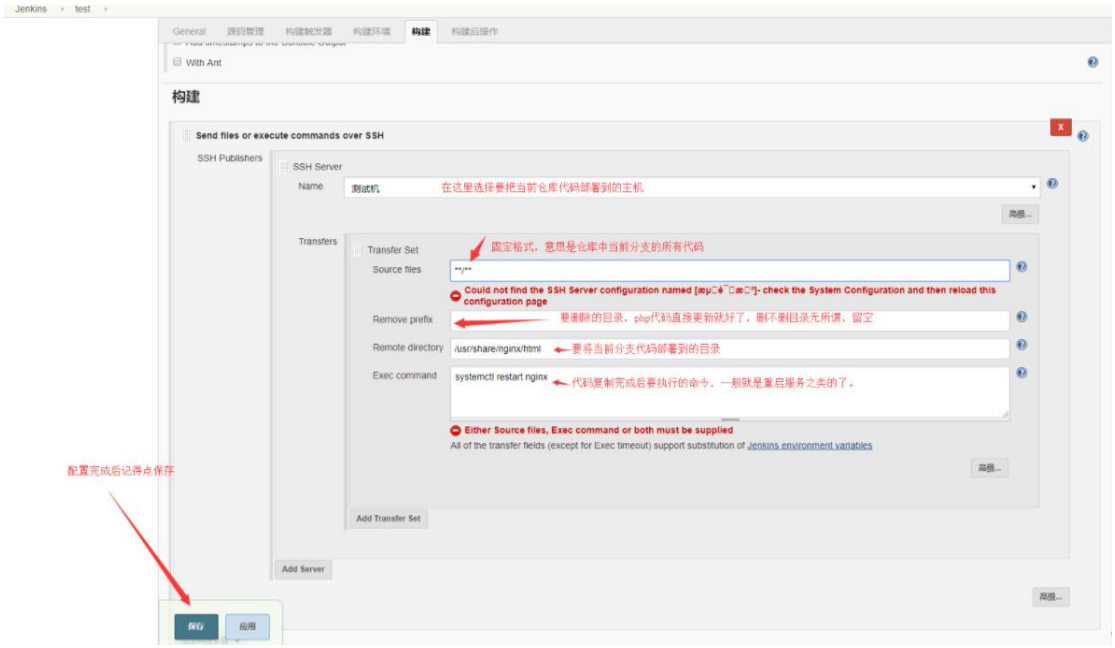

[root@jenkins ~]# cat /root/.ssh/id_rsa.pub
[root@jenkins ~]# cat /root/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
-----END RSA PRIVATE KEY-----
[root@jenkins ~]#
这个公钥文件是复制到gitlab服务器指定位置
这个私钥文件是复制到jenkins指定位置


Jenkins
test
General
源码管理
构建触发器
构建环境
构建
构建后操作
关闭构建
在必要的时候并发构建
高级...
源码管理
无
Git
Repositories
Repository URL
git@139.9.91.82:root/test.git
Credentials
root
添加
- 无 -
root
Name
Refspec
Add Repository
Branches to build
Branch Specifier (blank for 'any')
*/master
Add Branch
源码库浏览器
保存
应用
新增
添加完后，选择 root 或 git 用户进行认证
没有任何提示表示成功

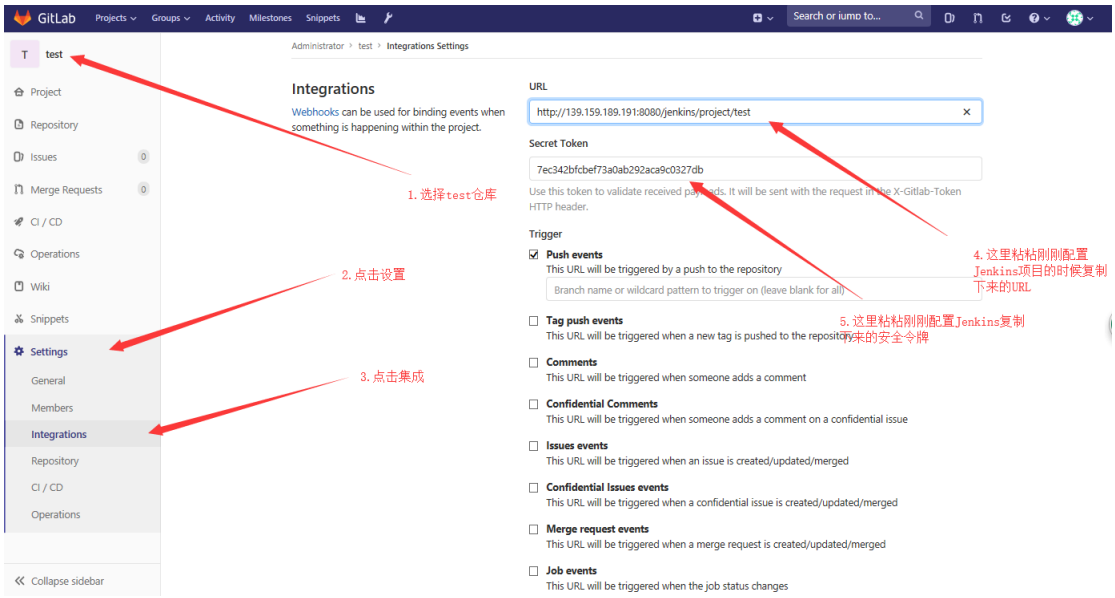
构建触发器

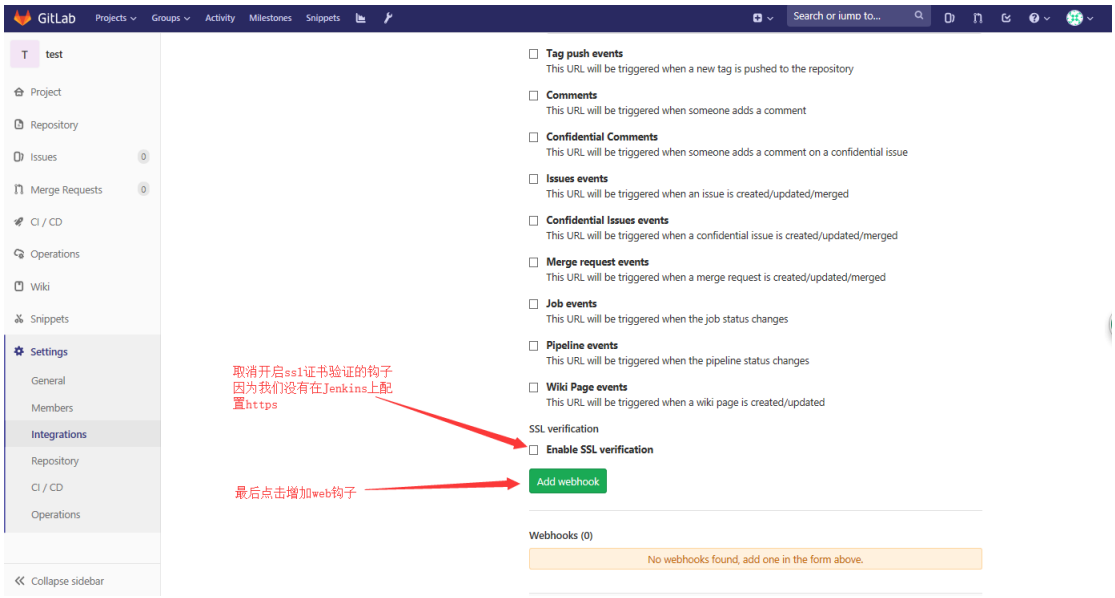

General
源码管理
构建触发器
构建环境
构建
构建后操作
构建触发器
触发远程构建 (例如,使用脚本)
其他工程构建后触发
定时构建
Build when a change is pushed to GitLab. GitLab webhook URL: http://139.159.189.191:8080/jenkins/project/test
复制这一段内容，保存到一个文件里
等下构建gitlab要用到
Enabled GitLab triggers
Push Events
Opened Merge Request Events
Accepted Merge Request Events
Closed Merge Request Events
Rebuild open Merge Requests
Never
Approved Merge Requests (EE-only)
Comments
Comment (regex) for triggering a build
Jenkins please retry a build
高级...
然后点高级
GitHub hook trigger for GITScm polling
轮询 SCM
保存
应用
这里一定要勾上

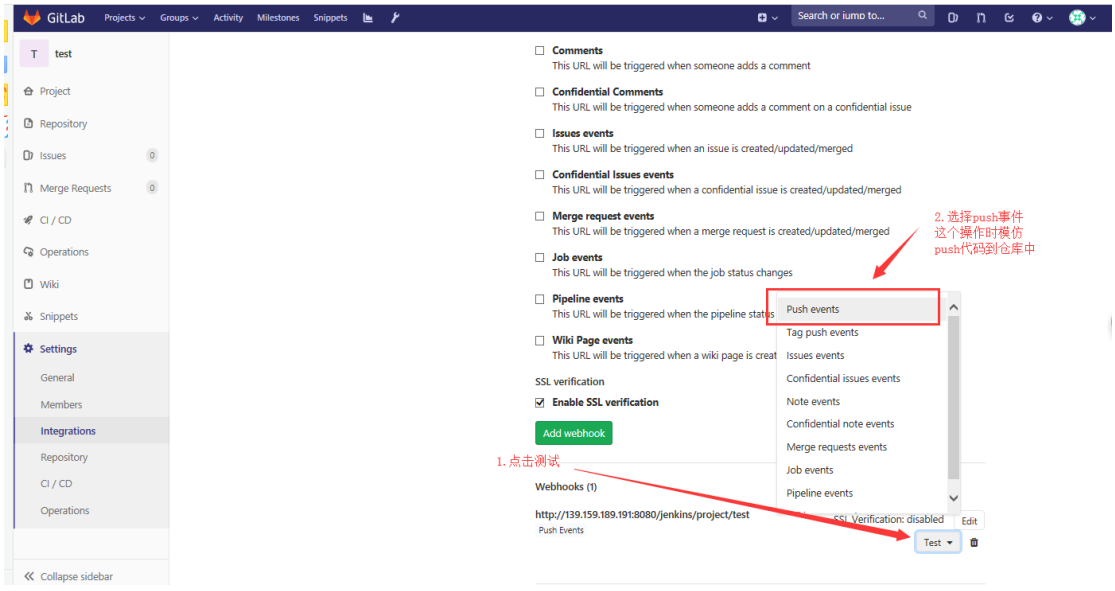

Jenkins
test
General
源码管理
构建触发器
构建环境
构建
构建后操作
Comments
Comment (regex) for triggering a build
Jenkins please retry a bui
Enable [ci-skip]
Ignore WIP Merge Requests
Set build description to build cause (eg. Merge request or Git Push)
Build on successful pipeline events
Pending build name for pipeline
Cancel pending merge request builds on update
Allowed branches
Allow all branches to trigger this job
Filter branches by name
Filter branches by regex
选择要自动部署的分支
这里使用正则表达式来匹配
Source Branch Regex
Target Branch Regex
点击Generate按钮
生成gitlab webhook的安全令牌
然后复制这段代码，待会配置gitlab会用到
Filter merge request by label
Secret token
7ec342bfcbef73a0ab292aca9c0327db
Generate
Clear
保存
应用

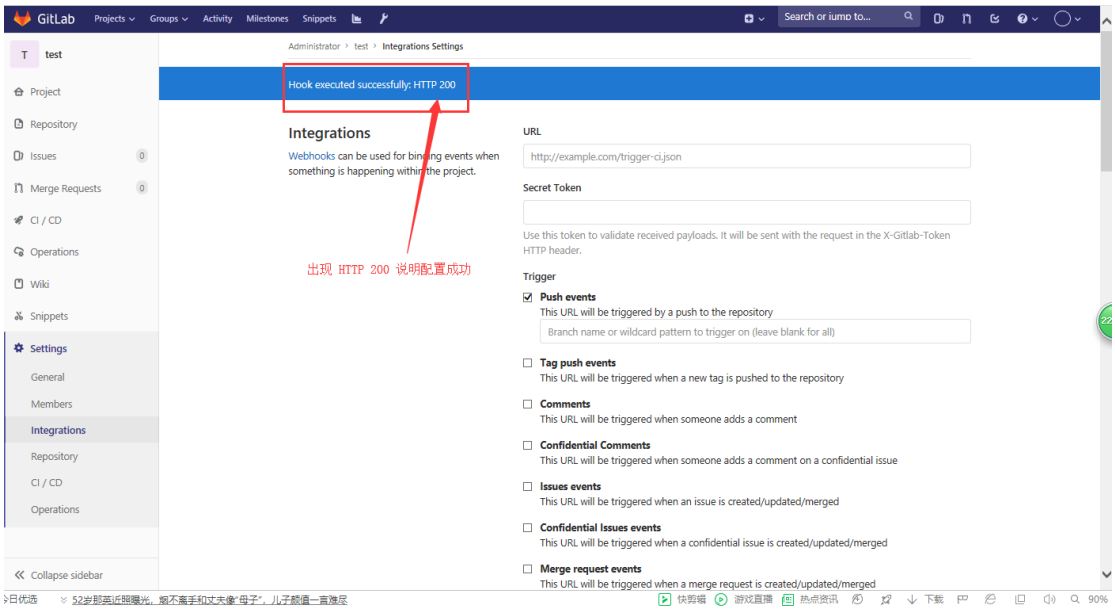


#配置 gitlab，当有代码提交时，触发 jenkins 的部署操作




GitLab
Projects
Groups
Activity
Milestones
Snippets
Search or jump to...
test
Project
Repository
Issues
Merge Requests
CI / CD
Operations
Wiki
Snippets
Settings
General
Members
Integrations
Repository
CI / CD
Operations
Collapse sidebar
Tag push events
This URL will be triggered when a new tag is pushed to the repository
Comments
This URL will be triggered when someone adds a comment
Confidential Comments
This URL will be triggered when someone adds a comment on a confidential issue
Issues events
This URL will be triggered when an issue is created/updated/merged
Confidential Issues events
This URL will be triggered when a confidential issue is created/updated/merged
Merge request events
This URL will be triggered when a merge request is created/updated/merged
Job events
This URL will be triggered when the job status changes
Pipeline events
This URL will be triggered when the pipeline status changes
Wiki Page events
This URL will be triggered when a wiki page is created/updated
SSL verification
Enable SSL verification
Add webhook
Webhooks (0)
No webhooks found, add one in the form above.
取消开启ssl证书验证的钩子
因为我们没有在Jenkins上配置https
最后点击增加web钩子

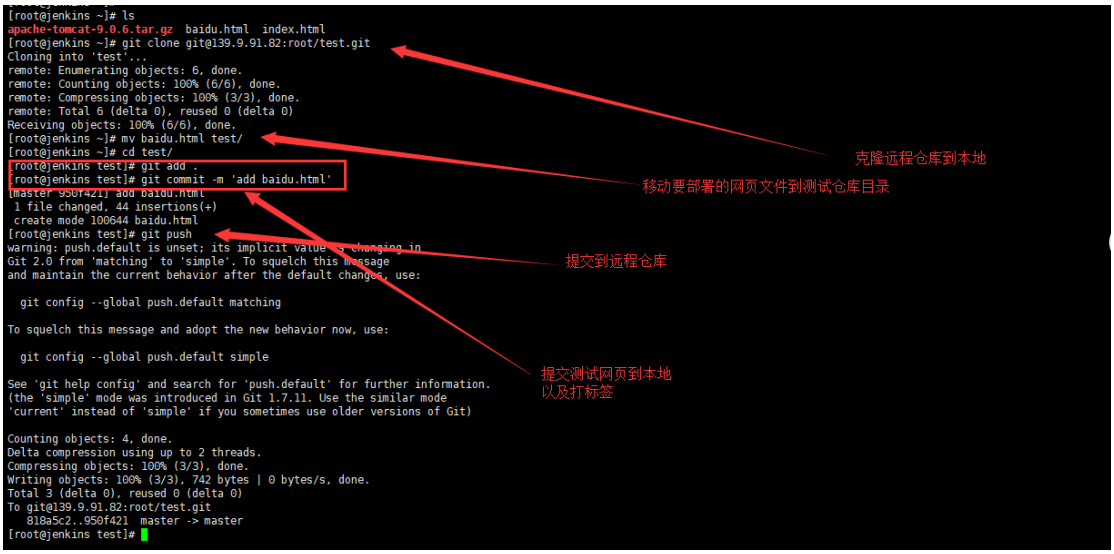

GitLab
Projects
Groups
Activity
Milestones
Snippets
Search or jump to...
test
Project
Repository
Issues
Merge Requests
CI / CD
Operations
Wiki
Snippets
Settings
General
Members
Integrations
Repository
CI / CD
Operations
Collapse sidebar
Comments
This URL will be triggered when someone adds a comment
Confidential Comments
This URL will be triggered when someone adds a comment on a confidential issue
Issues events
This URL will be triggered when an issue is created/updated/merged
Confidential Issues events
This URL will be triggered when a confidential issue is created/updated/merged
Merge request events
This URL will be triggered when a merge request is created/updated/merged
Job events
This URL will be triggered when the job status changes
Pipeline events
This URL will be triggered when the pipeline status
Wiki Page events
This URL will be triggered when a wiki page is creat
SSL verification
Enable SSL verification
Add webhook
Push events
Tag push events
Issues events
Confidential issues events
Note events
Confidential note events
Merge requests events
Job events
Pipeline events
2. 选择push事件
这个操作时模仿
push代码到仓库中
1. 点击测试
Webhooks (1)
http://139.159.189.191:8080/jenkins/project/test
Push Events
SSL Verification: disabled
Edit
Test


GitLab
Projects
Groups
Activity
Milestones
Snippets
Search or jump to...
test
Project
Repository
Issues
Merge Requests
CI / CD
Operations
Wiki
Snippets
Settings
General
Members
Integrations
Repository
CI / CD
Operations
Collapse sidebar
Administrator › test › Integrations Settings
Hook executed successfully: HTTP 200
Integrations
Webhooks can be used for binding events when something is happening within the project.
URL
http://example.com/trigger-ci.json
Secret Token
Use this token to validate received payloads. It will be sent with the request in the X-Gitlab-Token HTTP header.
Trigger
Push events
This URL will be triggered by a push to the repository
Branch name or wildcard pattern to trigger on (leave blank for all)
Tag push events
This URL will be triggered when a new tag is pushed to the repository
Comments
This URL will be triggered when someone adds a comment
Confidential Comments
This URL will be triggered when someone adds a comment on a confidential issue
Issues events
This URL will be triggered when an issue is created/updated/merged
Confidential Issues events
This URL will be triggered when a confidential issue is created/updated/merged
Merge request events
This URL will be triggered when a merge request is created/updated/merged
出现 HTTP 200 说明配置成功

测试提交代码是否会部署到远程主机

#再次克隆仓库

git clone git@139.9.91.82:root/test.git
cd test/
echo '测试 Jenkins' > index.html
git add .
git commit -m 'add index.html'
git push



#提交之后去 jenkins 页面看看

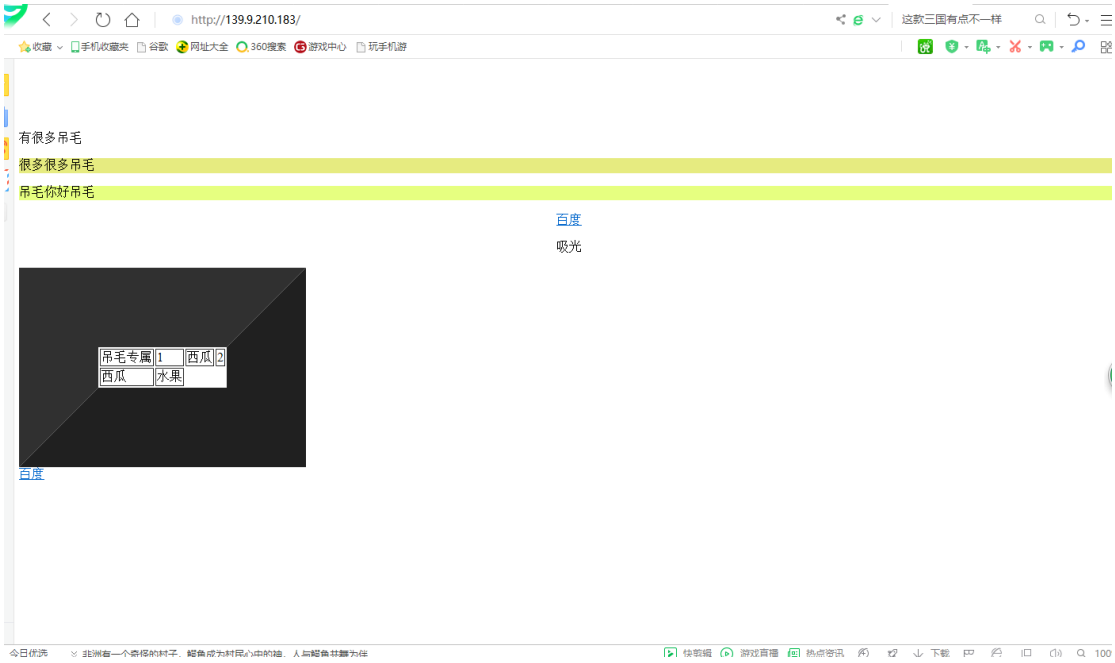

Jenkins
admin | 注销
Jenkins test
允许自动刷新
返回面板
状态
修改记录
工作空间
立即构建
删除 工程
配置
重命名
工程 test
添加说明
禁用项目
工作区
最新修改记录
相关链接
最近一次构建(#3),11 分之前
最近稳定构建(#3),11 分之前
最近成功的构建(#3),11 分之前
最近完成的构建(#3),11 分之前
Build History
构建历史
find
#3 2019-7-14 下午8:57
Started by GitLab push by Administrator
#2 2019-7-14 下午8:45
Started by GitLab push by Administrator
#1 2019-7-14 下午8:39
Started by GitLab push by Administrator
RSS 全部
RSS 失败
刚刚我提交了一次网页测试文件
已经部署到web服务器上了


Jenkins
Jenkins test #2
返回到工程
状态集
变更记录
控制台输出
文本方式查看
编辑编译信息
删除本次生成
轮询日志
Git Build Data
No Tags
前一次构建
控制台输出
Started by GitLab push by Administrator
Building in workspace /usr/share/tomcat/.jenkins/workspace/test
> git rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url git@192.168.1.2:root/test.git # timeout=10
Fetching upstream changes from git@192.168.1.2:root/test.git
> git --version # timeout=10
using GIT_SSH to set credentials
> git fetch --tags --progress git@192.168.1.2:root/test.git +refs/heads/*:refs/remotes/origin/*
> git rev-parse remotes/origin/master^{commit} # timeout=10
> git branch -a -v --no-abbrev --contains e5a0cfe2effad9921c8f72a017b65899b542e807 # timeout=10
Checking out Revision e5a0cfe2effad9921c8f72a017b65899b542e807 (origin/master)
> git config core.sparsecheckout # timeout=10
> git checkout -f e5a0cfe2effad9921c8f72a017b65899b542e807
Commit message: "add index.html"
> git rev-list --no-walk a12fa48054519a923abacc3ad514d482104a5581 # timeout=10
SSH: Connecting from host [jenkins]
SSH: Connecting with configuration [测试机] ...
SSH: EXEC: STDOUT/STDERR from command [systemctl restart nginx] ...
SSH: EXEC: completed after 403 ms
SSH: Disconnecting configuration [测试机] ...
SSH: Transferred 3 file(s)
Build step 'Send files or execute commands over SSH' changed build result to SUCCESS
Finished: SUCCESS
通过Jenkins构建的控制台可以看到,
部署的代码分支是master分支
部署完成之后也执行了重启nginx的命令


Jenkins
admin | 注销
Jenkins test
关闭自动刷新
返回面板
状态
修改记录
工作空间
立即构建
删除 工程
配置
重命名
工程 test
点击立即构建
添加说明
禁用项目
工作区
最新修改记录
相关链接
最近一次构建(#5),4 分 14 秒之前
最近稳定构建(#5),4 分 14 秒之前
最近成功的构建(#5),4 分 14 秒之前
最近完成的构建(#5),4 分 14 秒之前
开启自动刷新
Build History
构建历史
find
#5 2019-7-14 下午10:03
#4 2019-7-14 下午9:37
Started by GitLab push by Administrator
#3 2019-7-14 下午8:57
Started by GitLab push by Administrator
#2 2019-7-14 下午8:45
Started by GitLab push by Administrator
#1 2019-7-14 下午8:39
Started by GitLab push by Administrator
RSS 全部
RSS 失败

查看 web 服务器

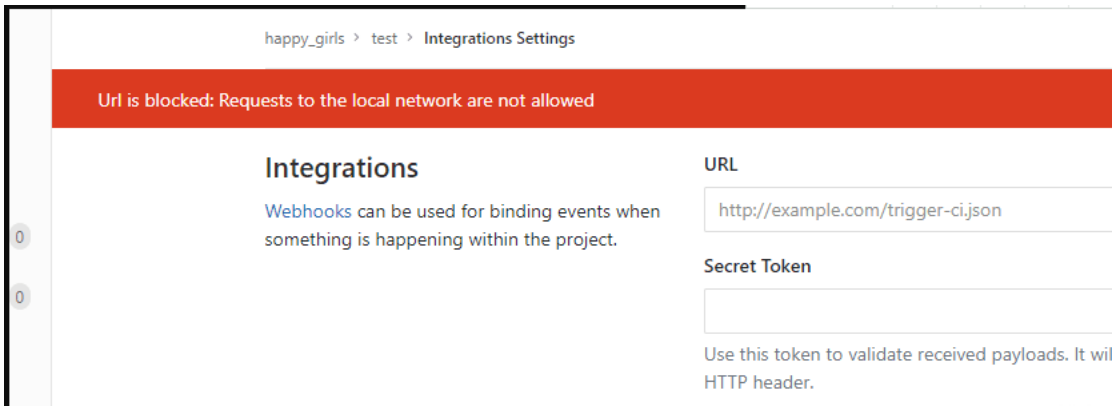
访问 web 测试页面



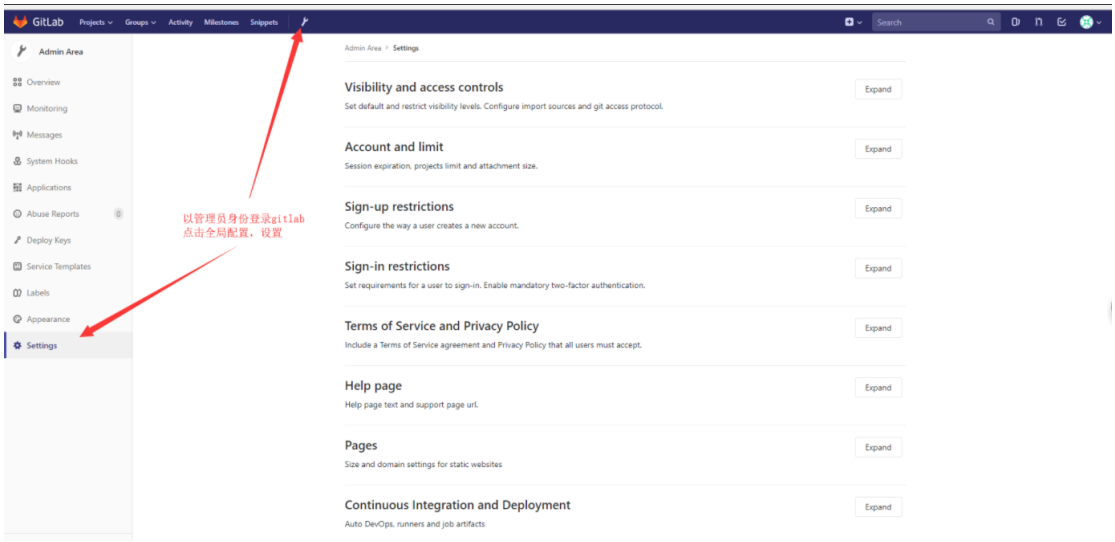
访问成功

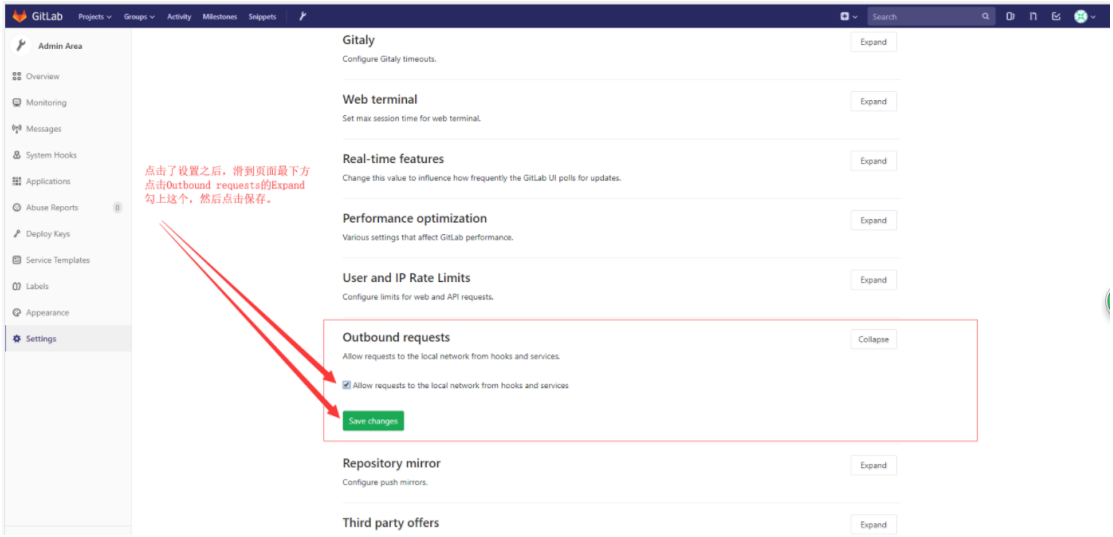
小坑：
错误提示：

#很多朋友使用最新版本的 gitlab 做自动部署时，在增加 web 钩子那一步，
#点击 test push events 时会报错：Url is blocked: Requests to the local network are not allowed



解决方法：
#这是因为新版的 gitlab 为了安全默认禁止了本地局域网地址调用 web hook
#我们在设置里允许就行，具体步骤如下：


GitLab
Projects
Groups
Activity
Milestones
Snippets
Search
Admin Area
Overview
Monitoring
Messages
System Hooks
Applications
Abuse Reports
Deploy Keys
Service Templates
Labels
Appearance
Settings
Admin Area › Settings
以管理员身份登录gitlab
点击全局配置，设置
Visibility and access controls
Set default and restrict visibility levels. Configure import sources and git access protocol.
Expand
Account and limit
Session expiration, projects limit and attachment size.
Expand
Sign-up restrictions
Configure the way a user creates a new account.
Expand
Sign-in restrictions
Set requirements for a user to sign-in. Enable mandatory two-factor authentication.
Expand
Terms of Service and Privacy Policy
Include a Terms of Service agreement and Privacy Policy that all users must accept.
Expand
Help page
Help page text and support page url.
Expand
Pages
Size and domain settings for static websites
Expand
Continuous Integration and Deployment
Auto DevOps, runners and job artifacts
Expand


GitLab
Projects
Groups
Activity
Milestones
Snippets
Search
Admin Area
Overview
Monitoring
Messages
System Hooks
Applications
Abuse Reports
Deploy Keys
Service Templates
Labels
Appearance
Settings
点击了设置之后，滑到页面最下方
点击Outbound requests的Expand
勾上这个，然后点击保存。
Gitaly
Configure Gitaly timeouts.
Expand
Web terminal
Set max session time for web terminal.
Expand
Real-time features
Change this value to influence how frequently the GitLab UI polls for updates.
Expand
Performance optimization
Various settings that affect GitLab performance.
Expand
User and IP Rate Limits
Configure limits for web and API requests.
Expand
Outbound requests
Allow requests to the local network from hooks and services.
Collapse
Allow requests to the local network from hooks and services
Save changes
Repository mirror
Configure push mirrors.
Expand
Third party offers
Expand